

#4

160

Application	108
Transactional	106
Middleware	104
Network OS	102
Hardware	100

Figure 1A

180

Java	C + +	Cobal	Small Talk	120
CORBA			124	
Networks OS			102	
Hardware			100	

Figure 1C

170

Java	118
Enterprise Java Beans	114
Network OS	102
Hardware	100

Figure 1B

190

Java	C + +	Cob al	Small Talk	120
Windows DNS (COM/ MTS) 134				
Network OS 102				
Hardware 100				

Figure 1D

195

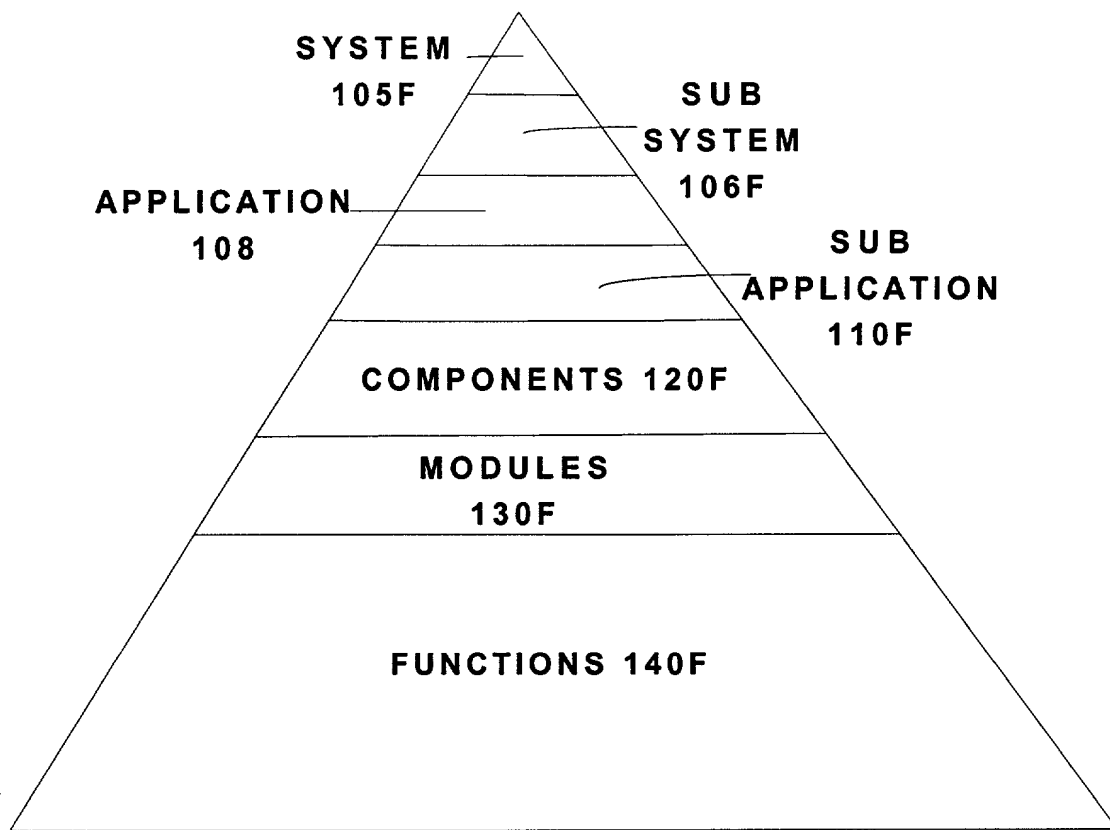
C	146
Tuxedo	144
Network OS	102
Hardware	100

Figure 1E

Patent # 6,544,600



100F



PRIOR ART

FIGURE 1F

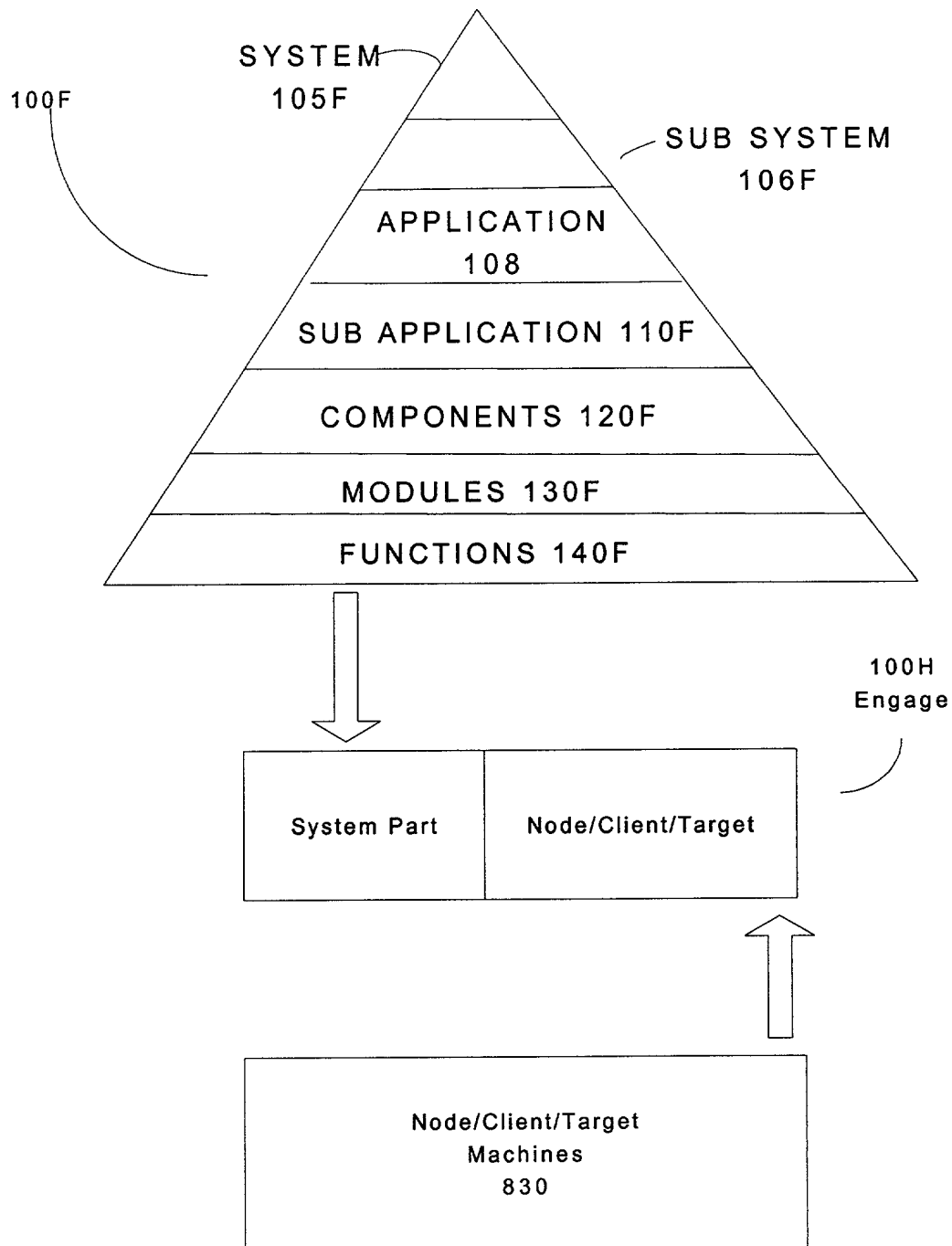


Figure 1G

100H

Part ID 120H	Target ID 130H	Engagement Pair 110H

•  
•  
•  
•

Figure 1H

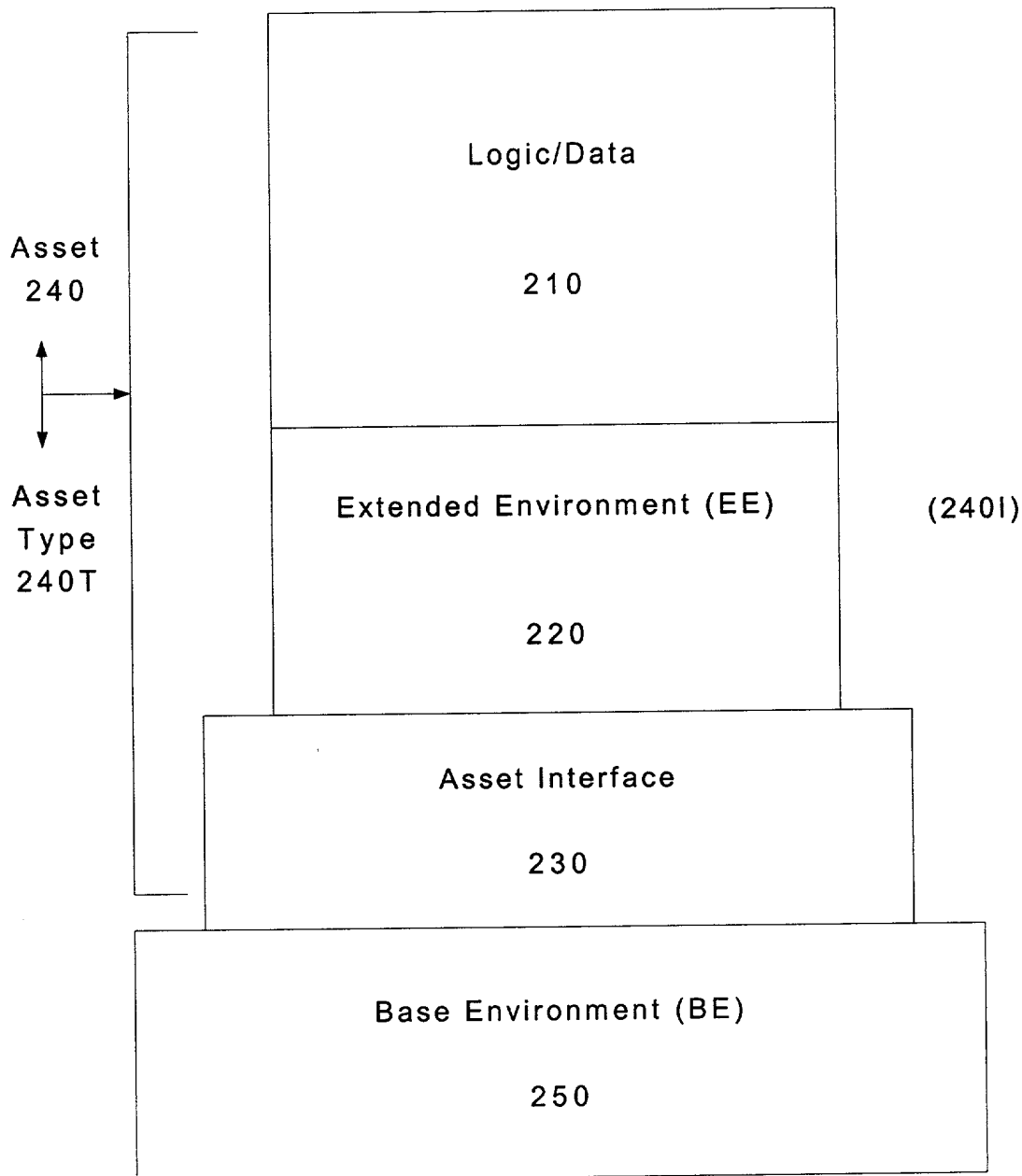


Figure 2

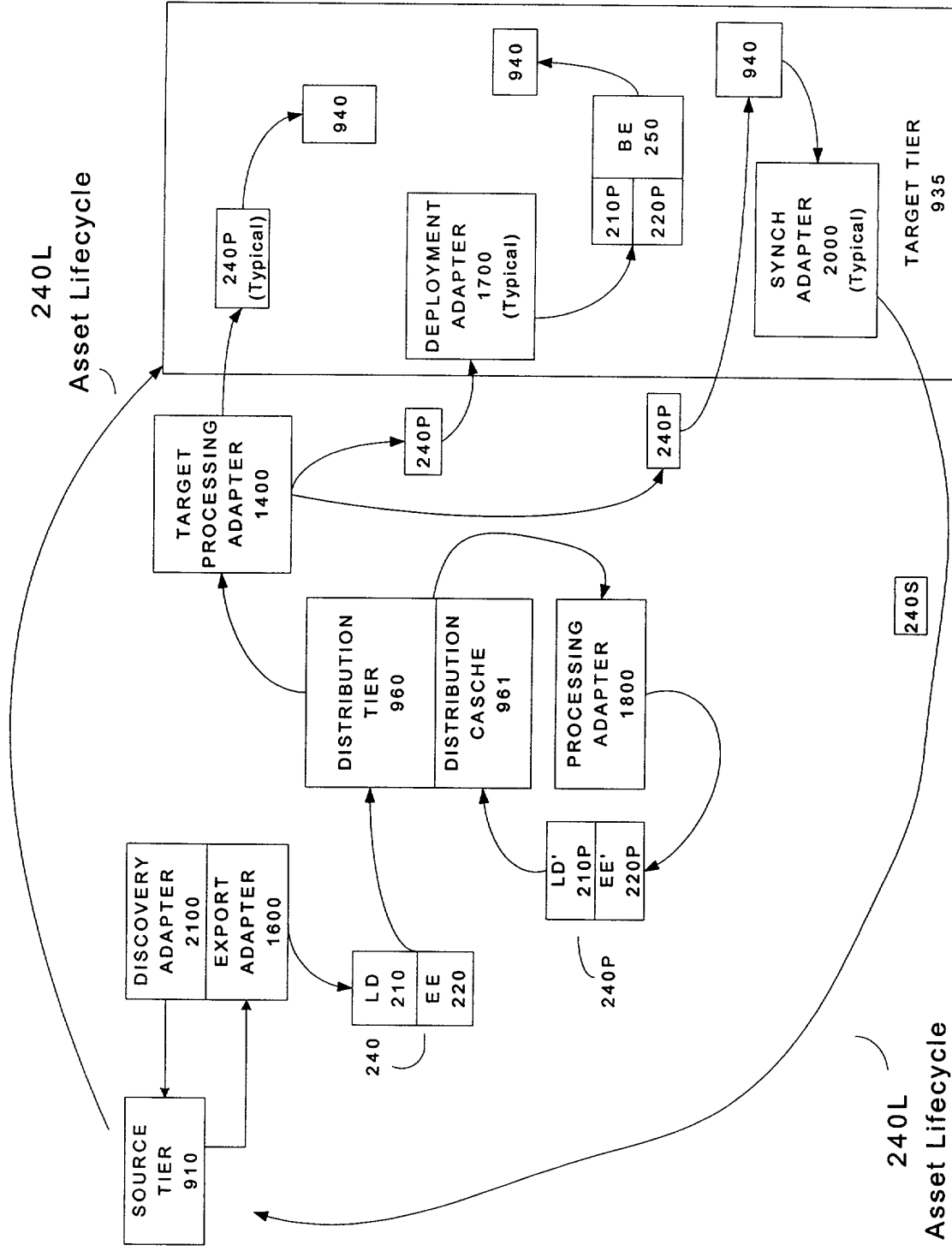
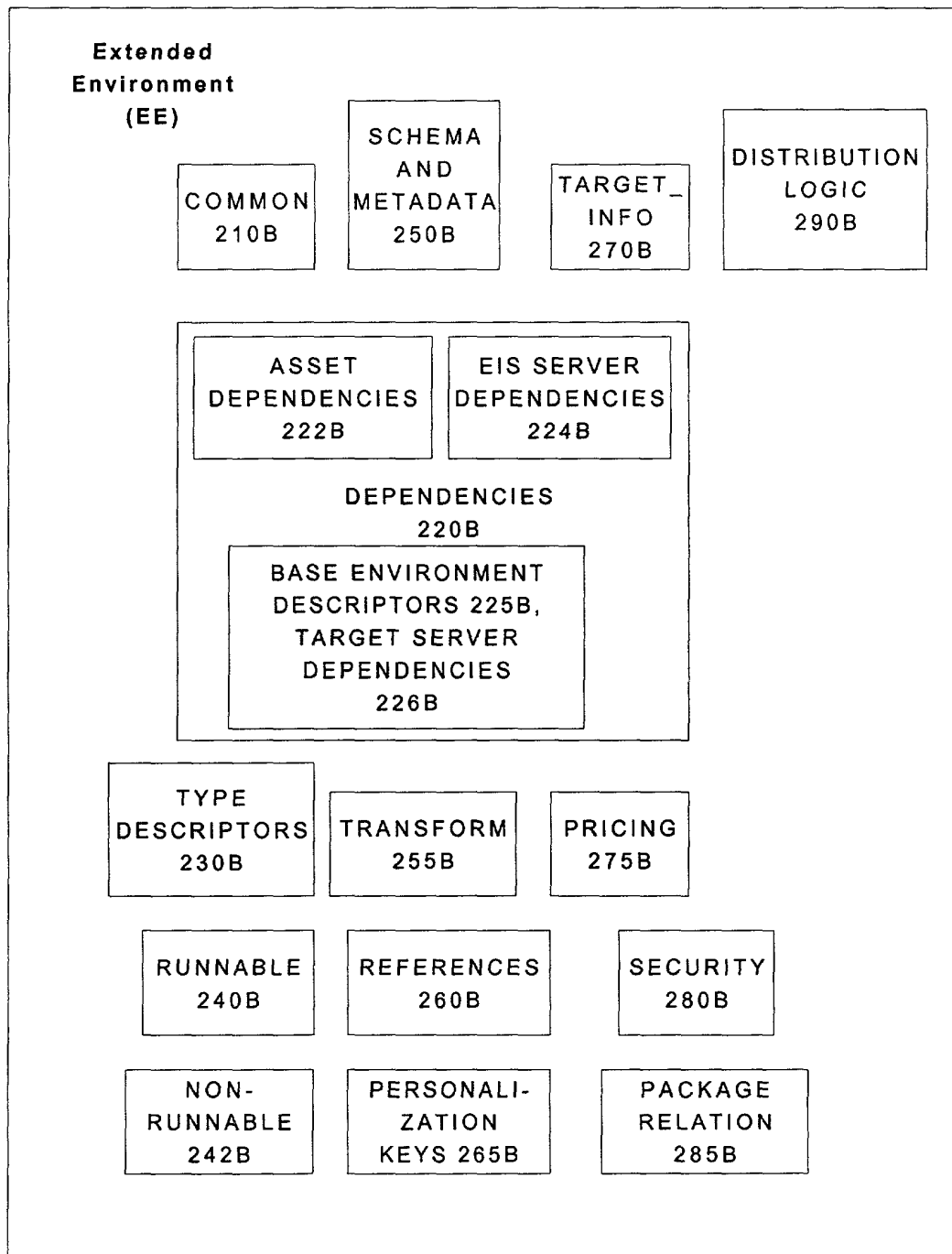


FIGURE 2A

**220**



**FIGURE 2B**

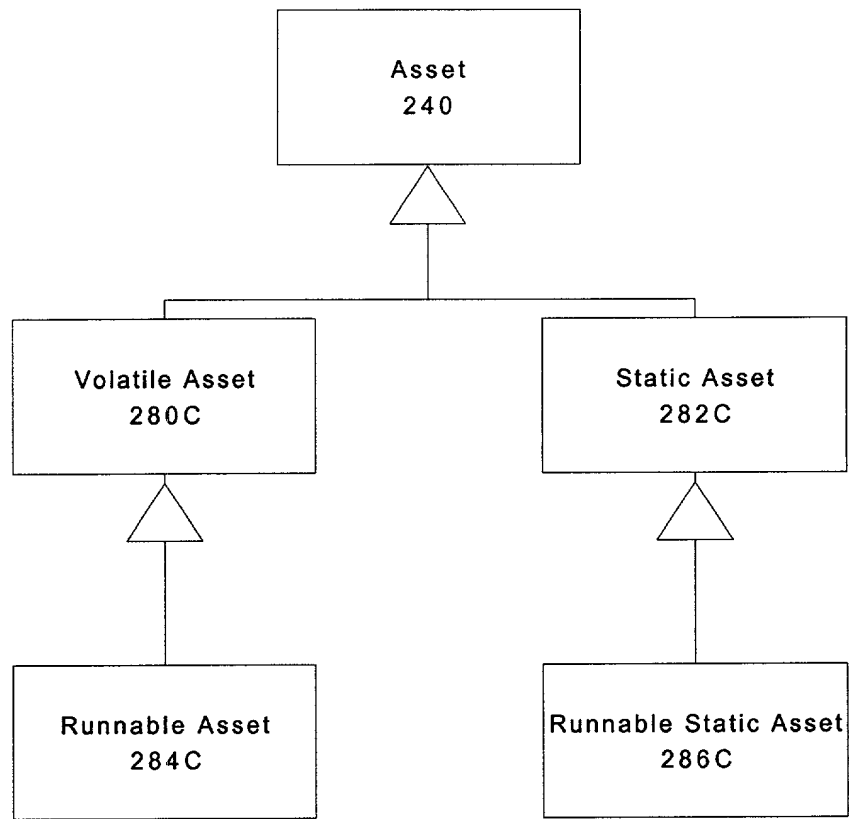
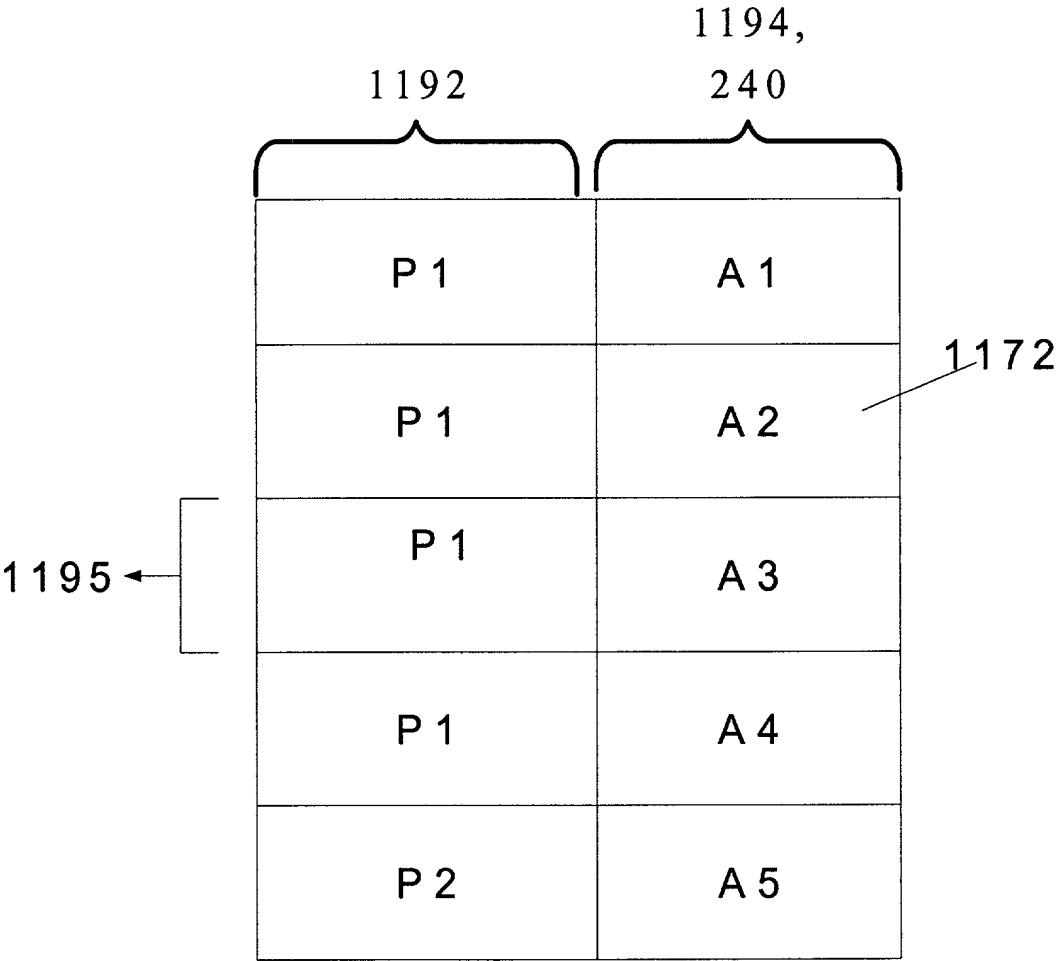


Figure 2C





1190



Package Content Data  
Structure

Figure 3

1194

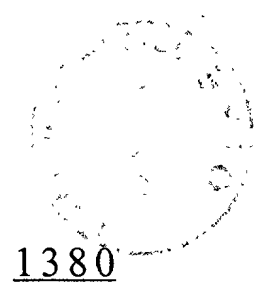
Asset ID	Location (Machine Location URL)	Name	Asset Type	Other (Optional)
1172	1174	1176	1178	1179
A 1			240T	
A 2				
A 3				

1175

Asset  
Definition Data  
Structure

Figure 4





Deployable Asset 1370  
Data Structure

Asset ID	Version
1372	1374

1375

Figure 5

Client  
Deployment  
Queue

Target/ Client ID 1382

1385

Figure 7

1390

Cient ID	Client Assets
1392	1394

Client Asset  
Table

Figure 6

Chen et al. 09/944,062

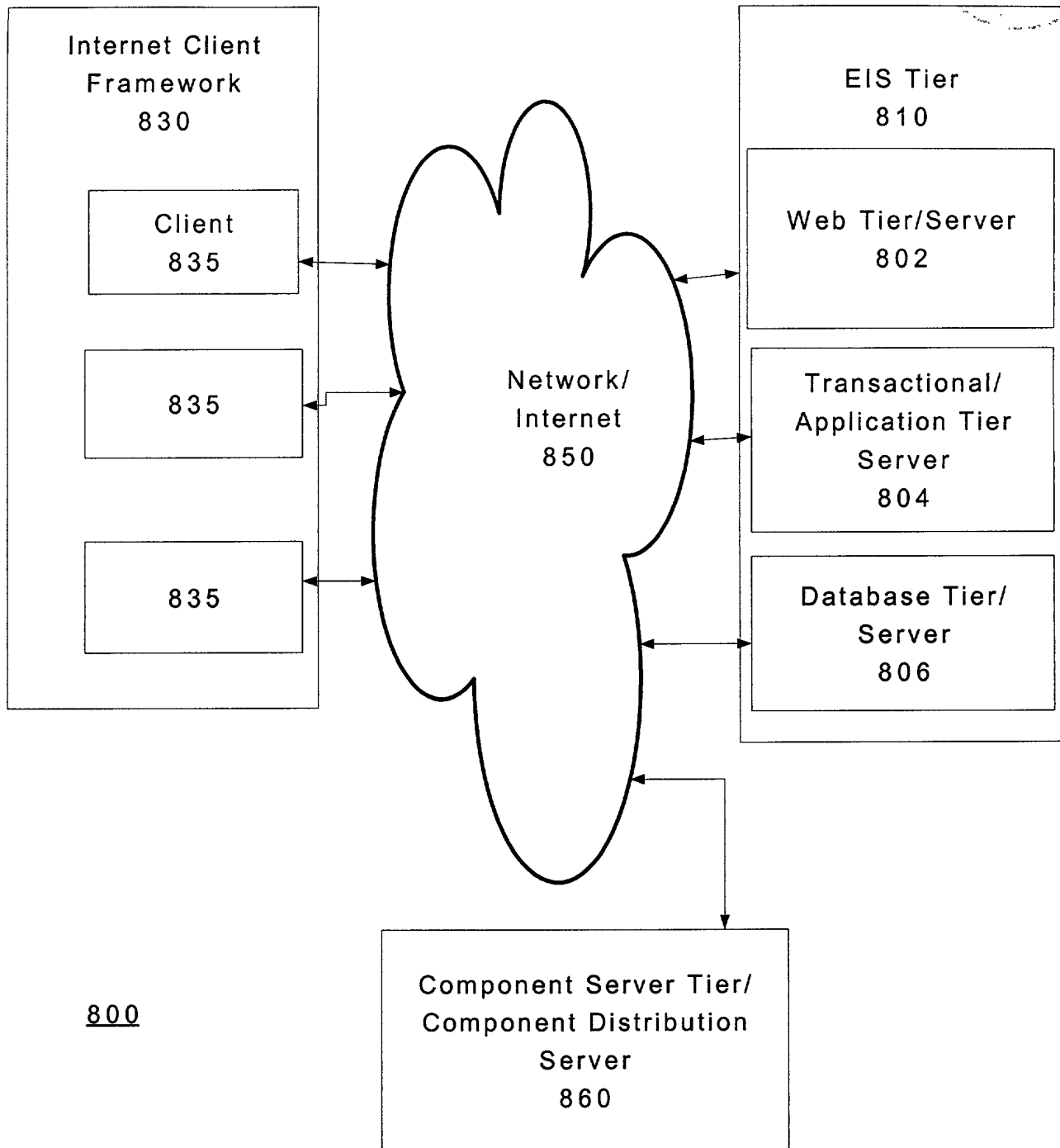


FIGURE 8

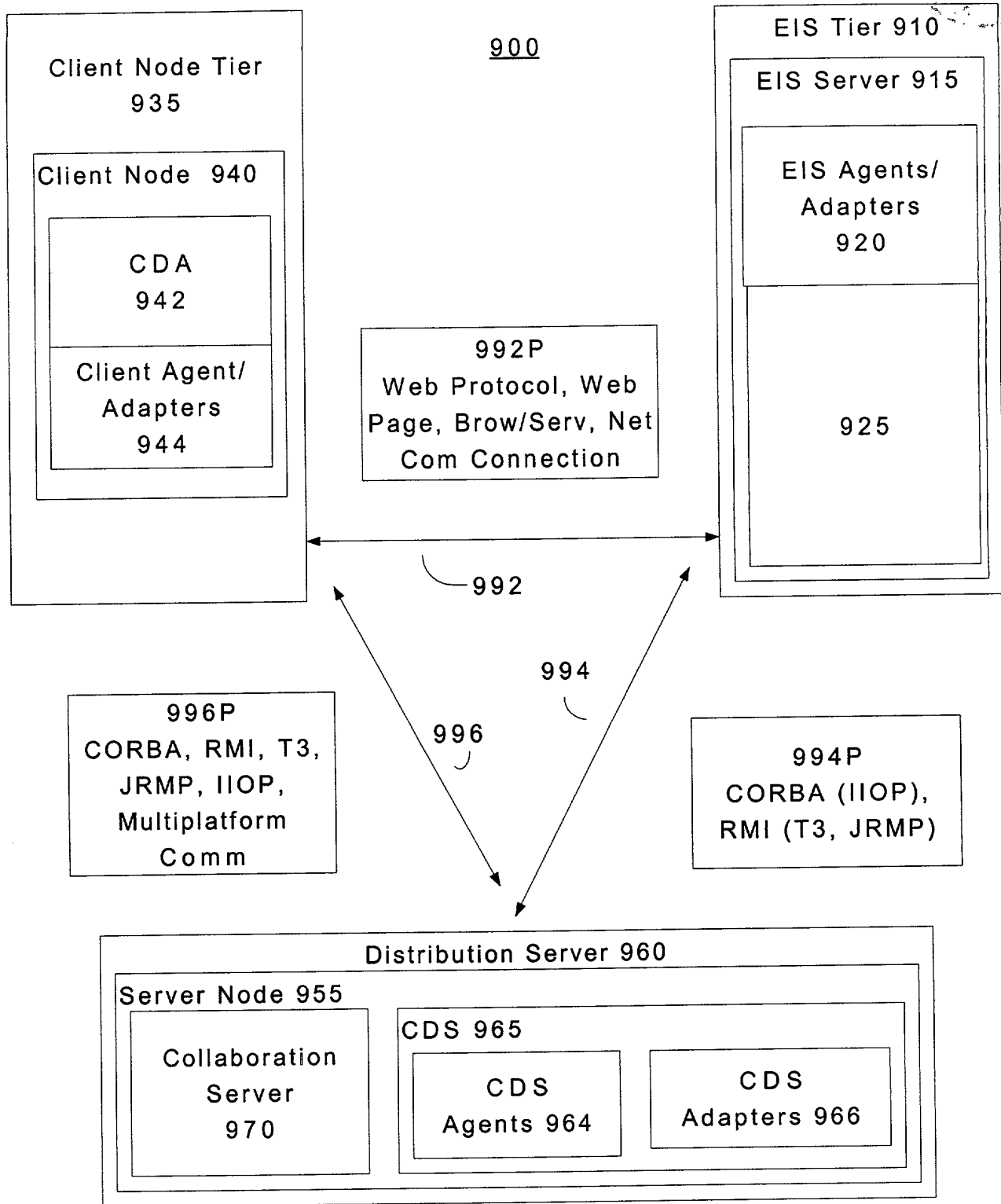
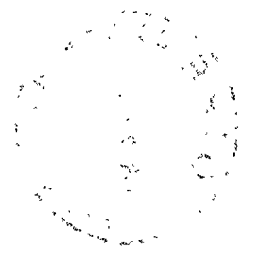


FIGURE 9



1000

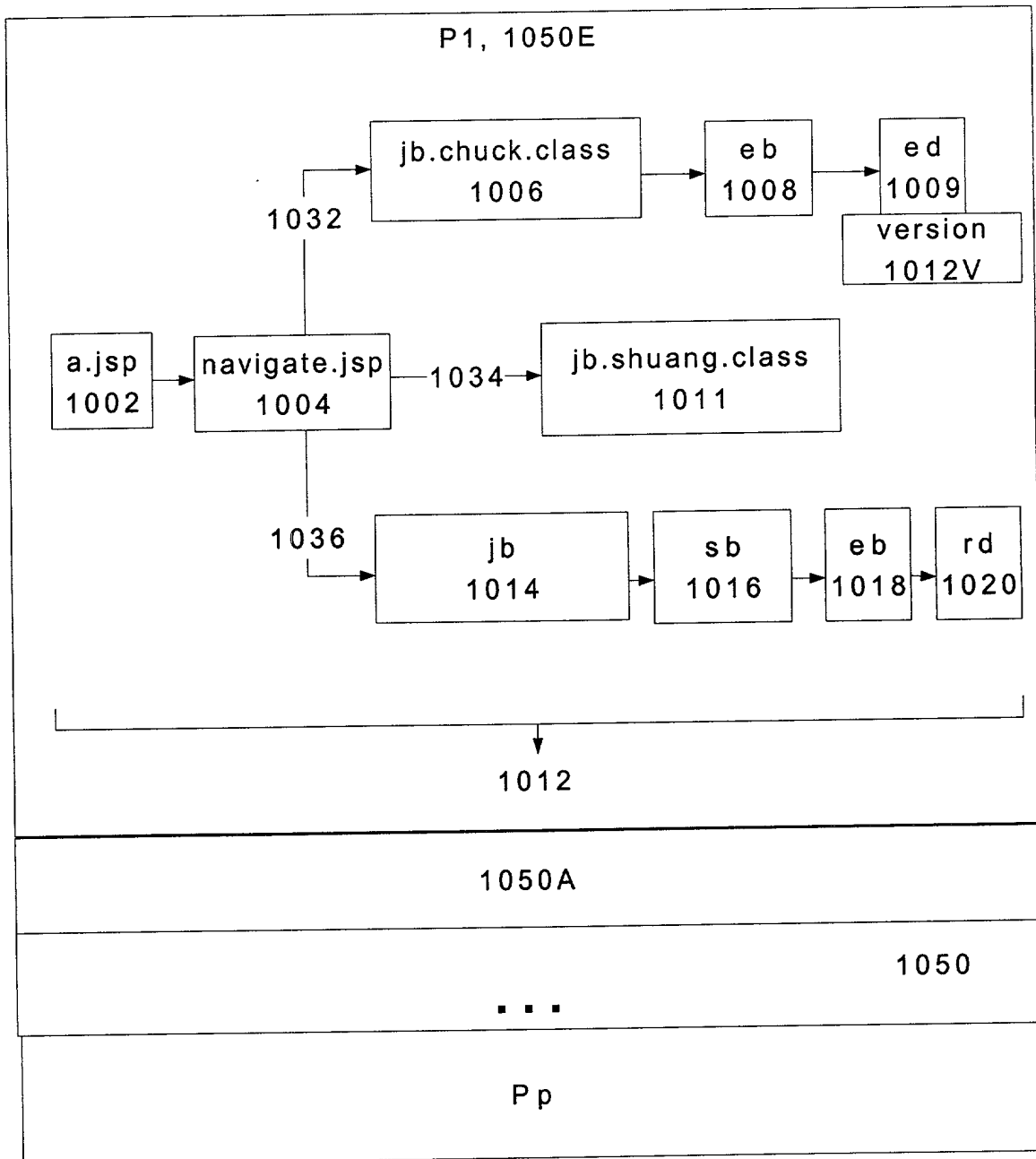


Figure 10

1100

Package ID 1110	1150						Location (e.g URL) 1120	Other 1163
	Immediate 1152	Delivery Start Time 1154	Delivery End Time 1156	Expire Time 1158	Remove Time 1160	Refresh Time 1162		
P 1								
P 1								
P 1								

1105

Package Definition Data Structure

Figure 11





1100A

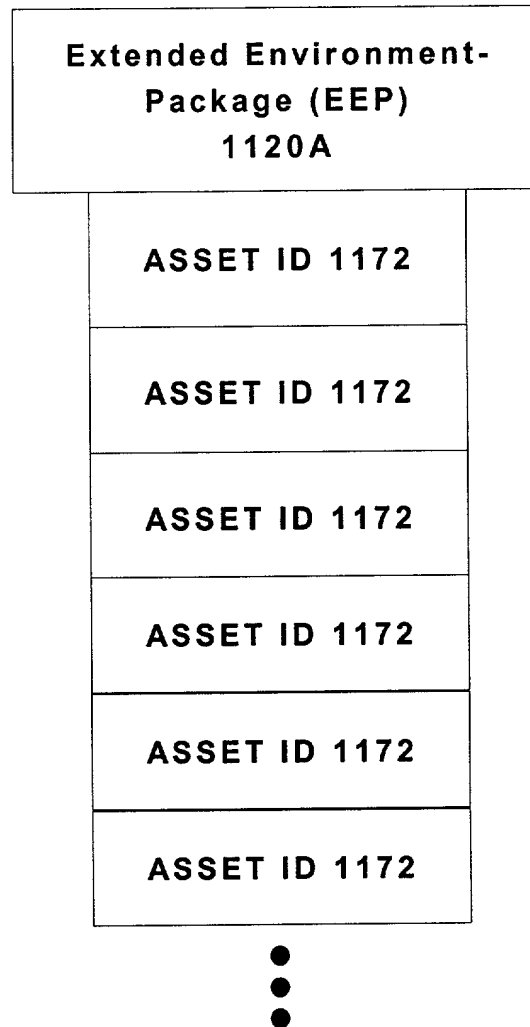
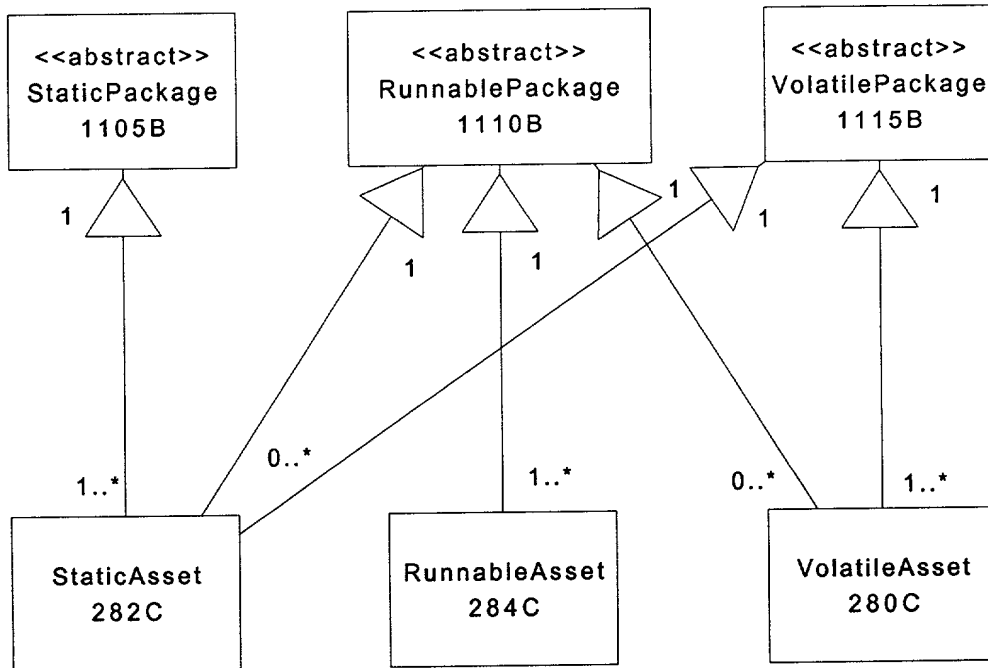


FIGURE 11A



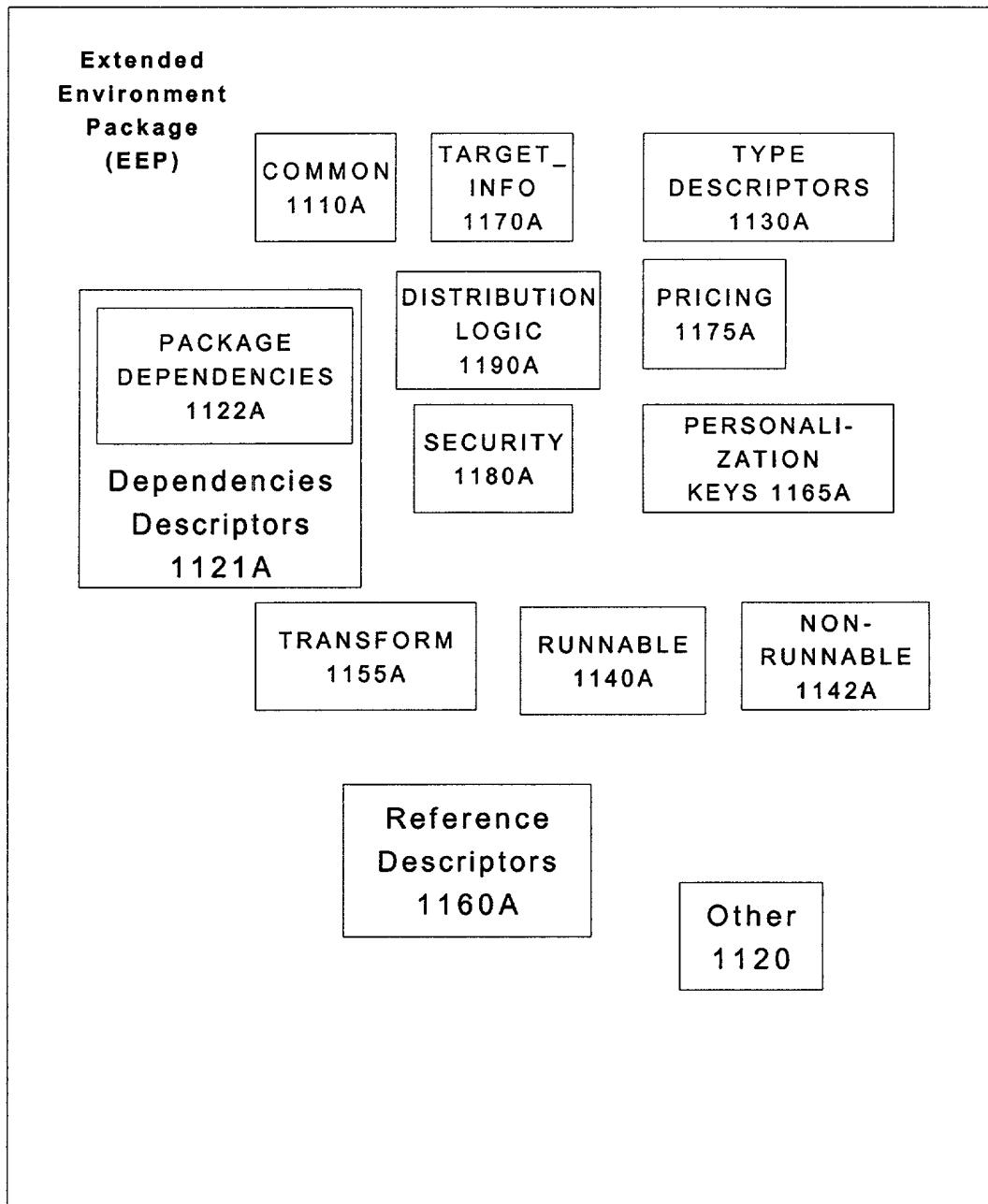


1100B

Figure 11B

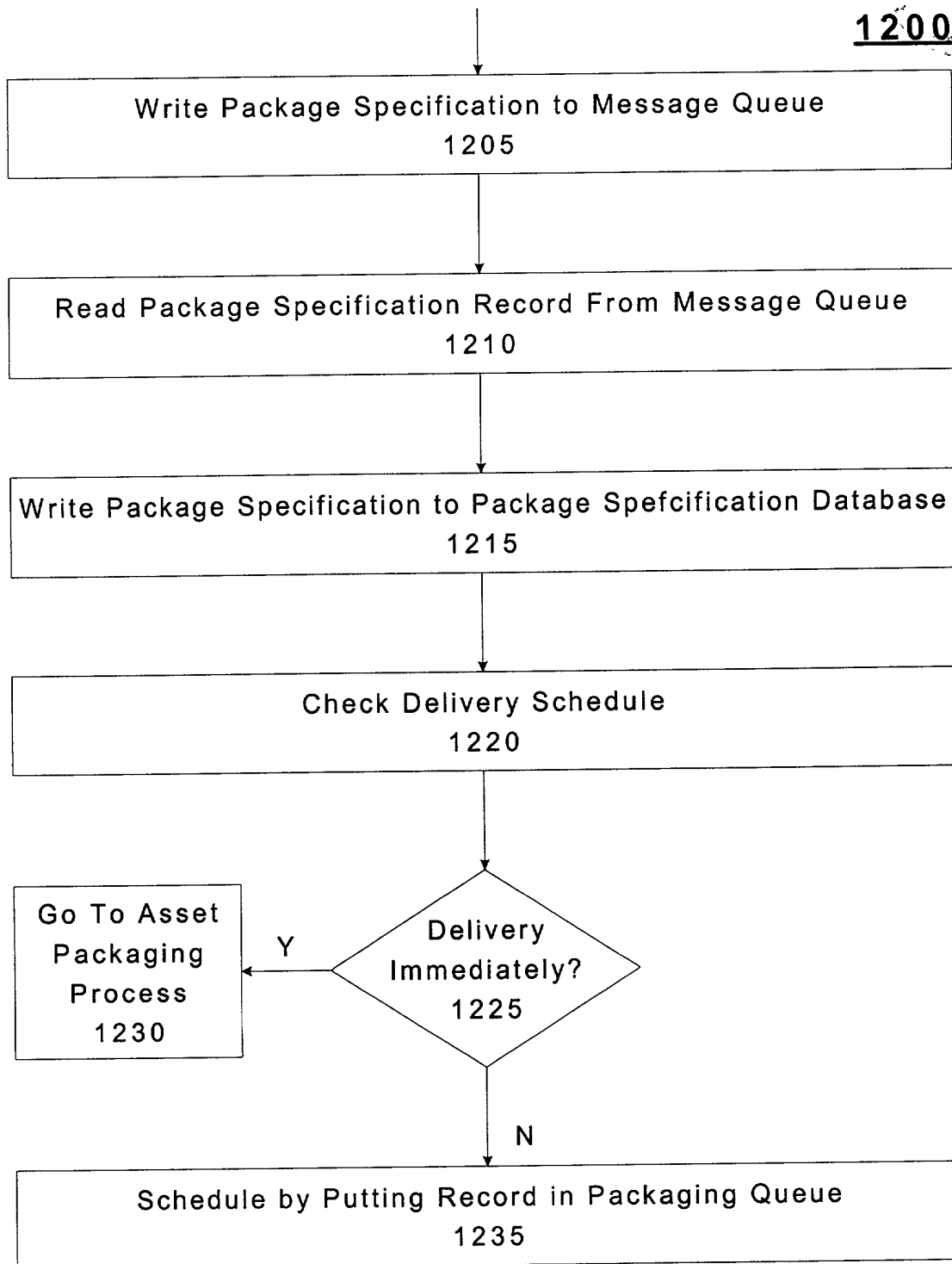


**1120A**



**FIGURE 11C**

1200



Package Specification Process

Figure 12

1250

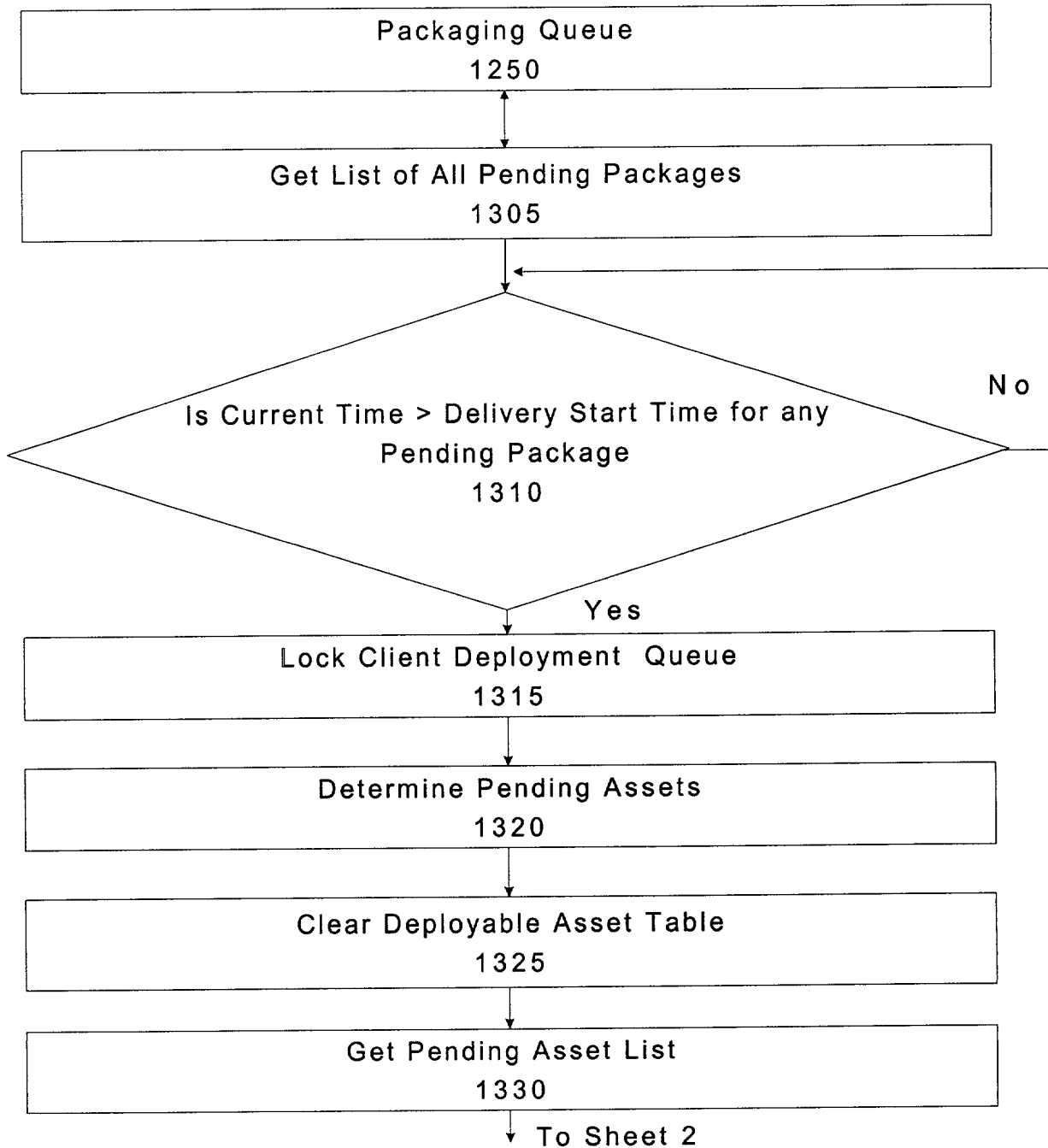
1255

Package ID 1252	Start Time 1254

Packaging  
Queue

Figure 12A

1300



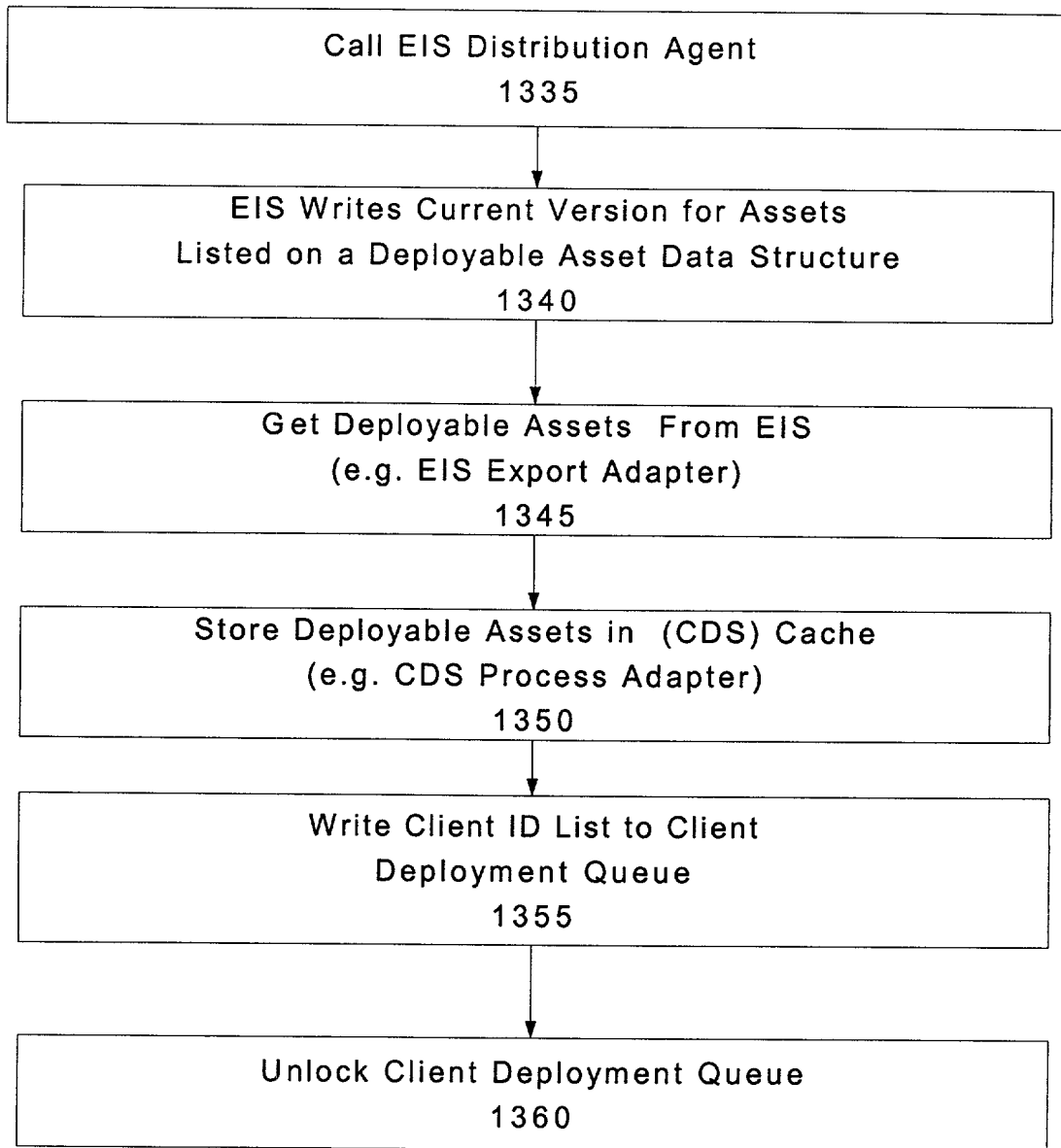
Asset Packaging Process

Figure 13 - Sheet 1



From Sheet 1

1300

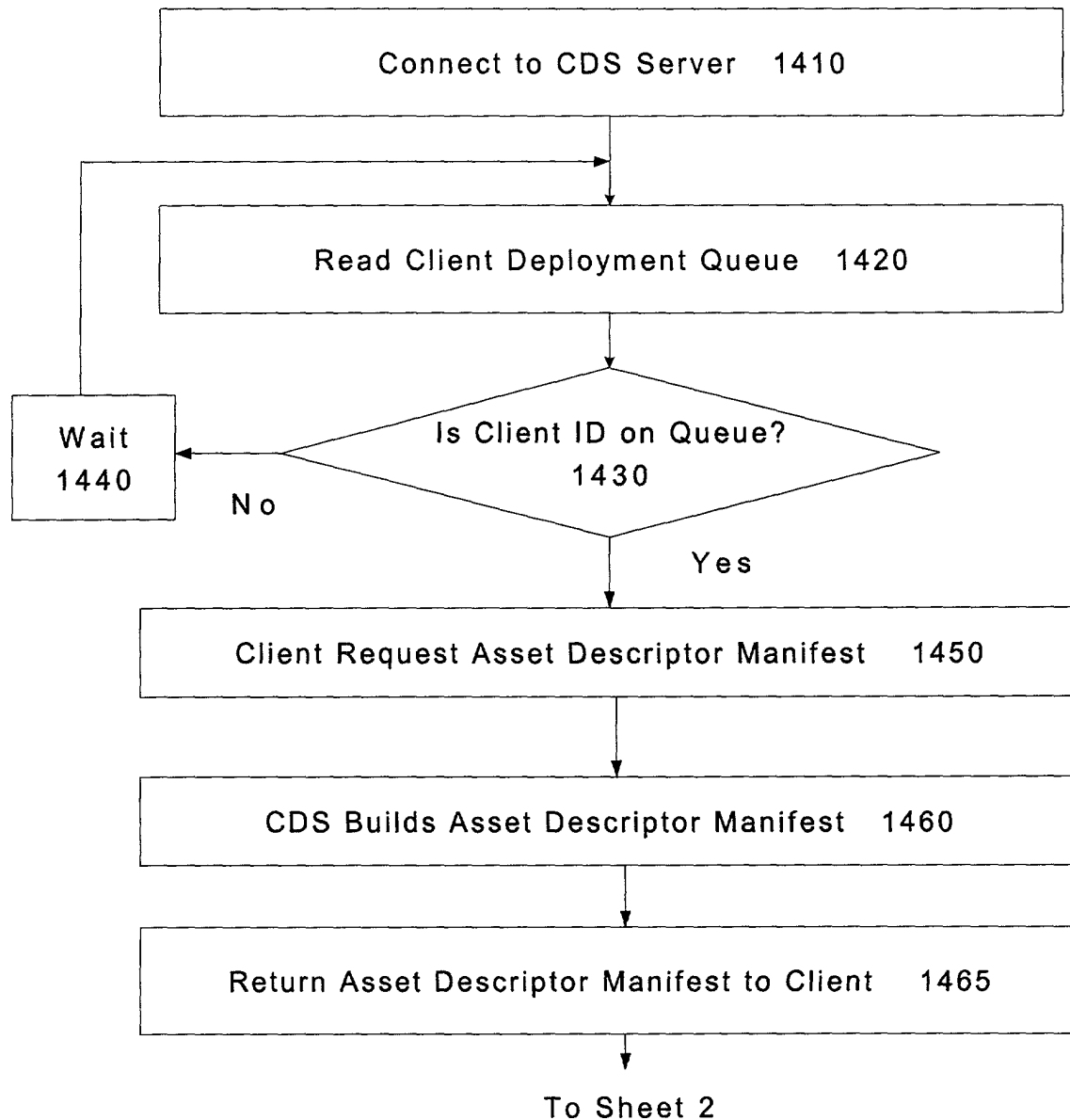


Asset Packaging Process

Figure 13 - Sheet 2



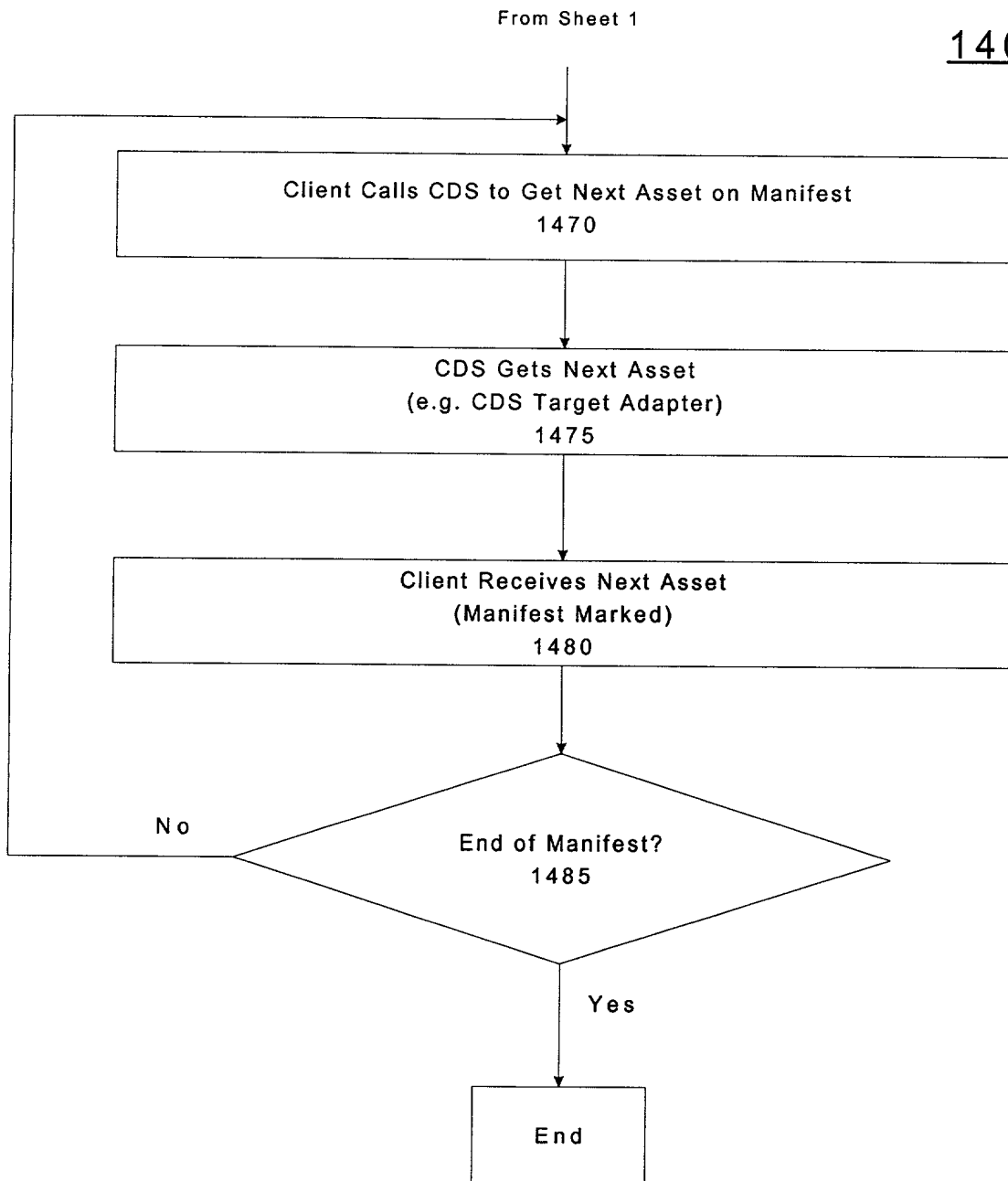
1400



Client Deployment Process

Figure 14 - Sheet 1

1400



Client Deployment Process

Figure 14 - Sheet 2

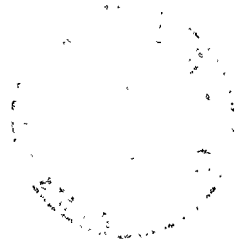


Client ID 1452				
Asset ID 1454	Offset 1456	Asset Type 1458 (Optional)	Cache Name 1478	Version (Timestamp) 1479

1453

Asset Descriptor  
Manifest Data  
Structure

Figure 14 A





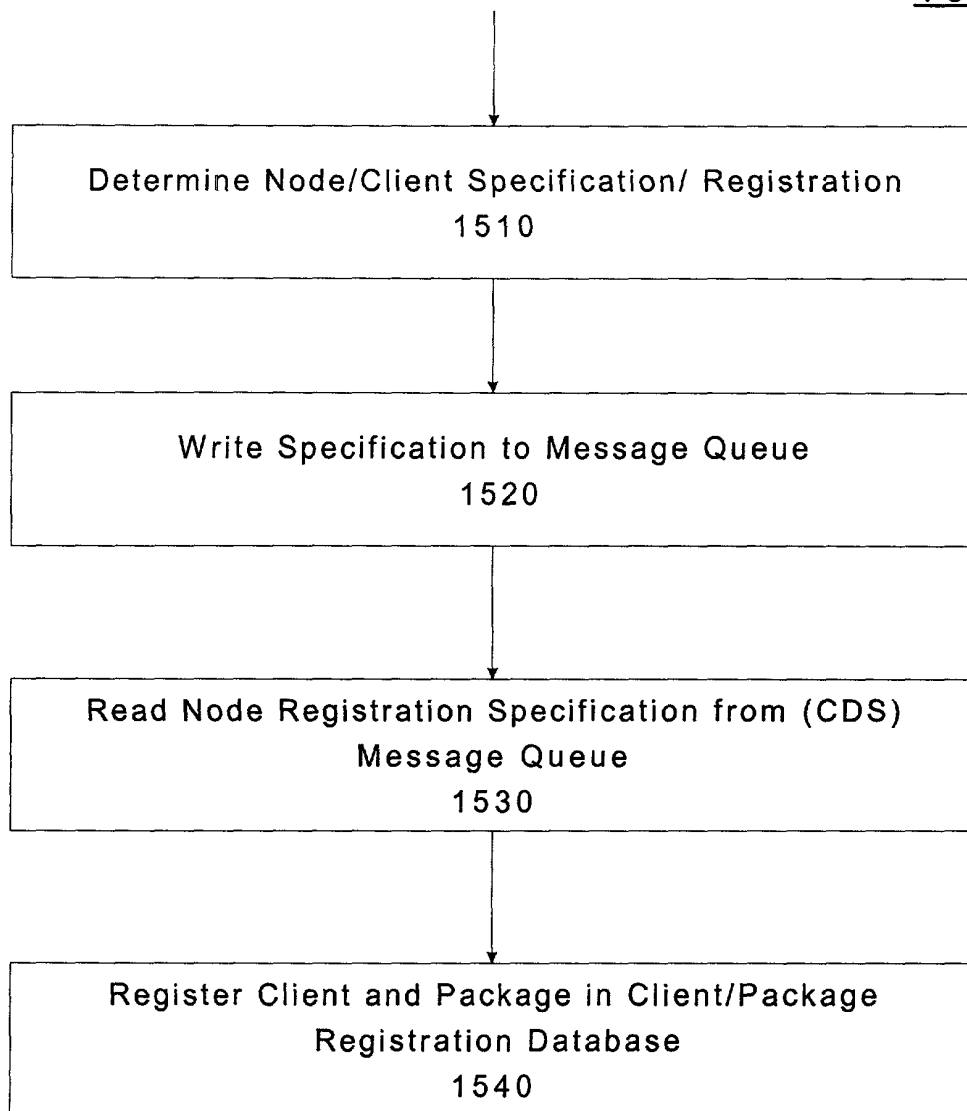
1495

<div>Client ID</div> <div>1452</div>	<div>Asset ID</div> <div>1454</div>	<div>Version</div> <div>(Timestamp)</div> <div>1479</div>
--------------------------------------	-------------------------------------	---

Client Asset Table

Figure 14B

1500



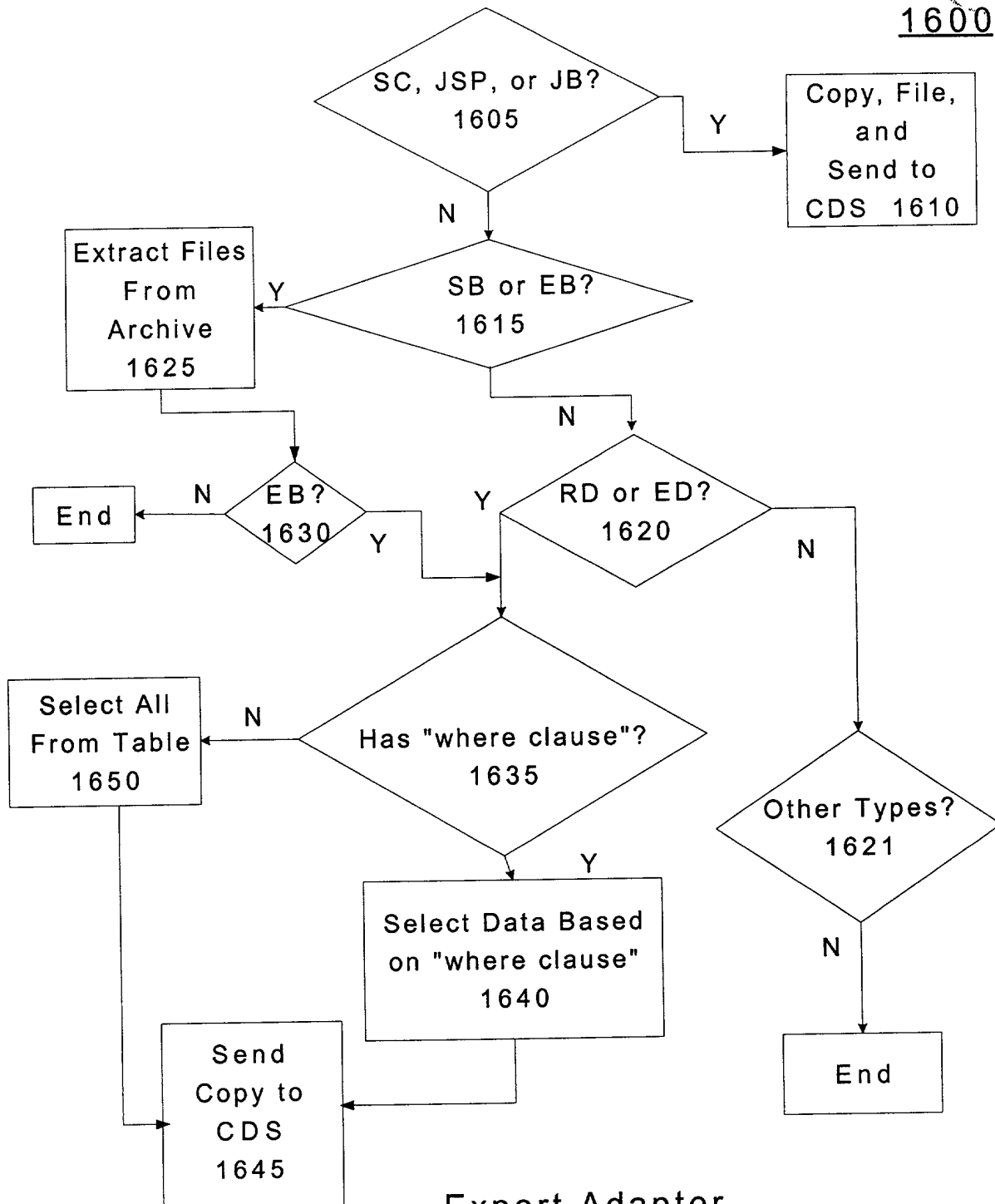
## Node Registration Process

Figure 15

Node ID 1524	Package ID 1526

Node Registration Specification

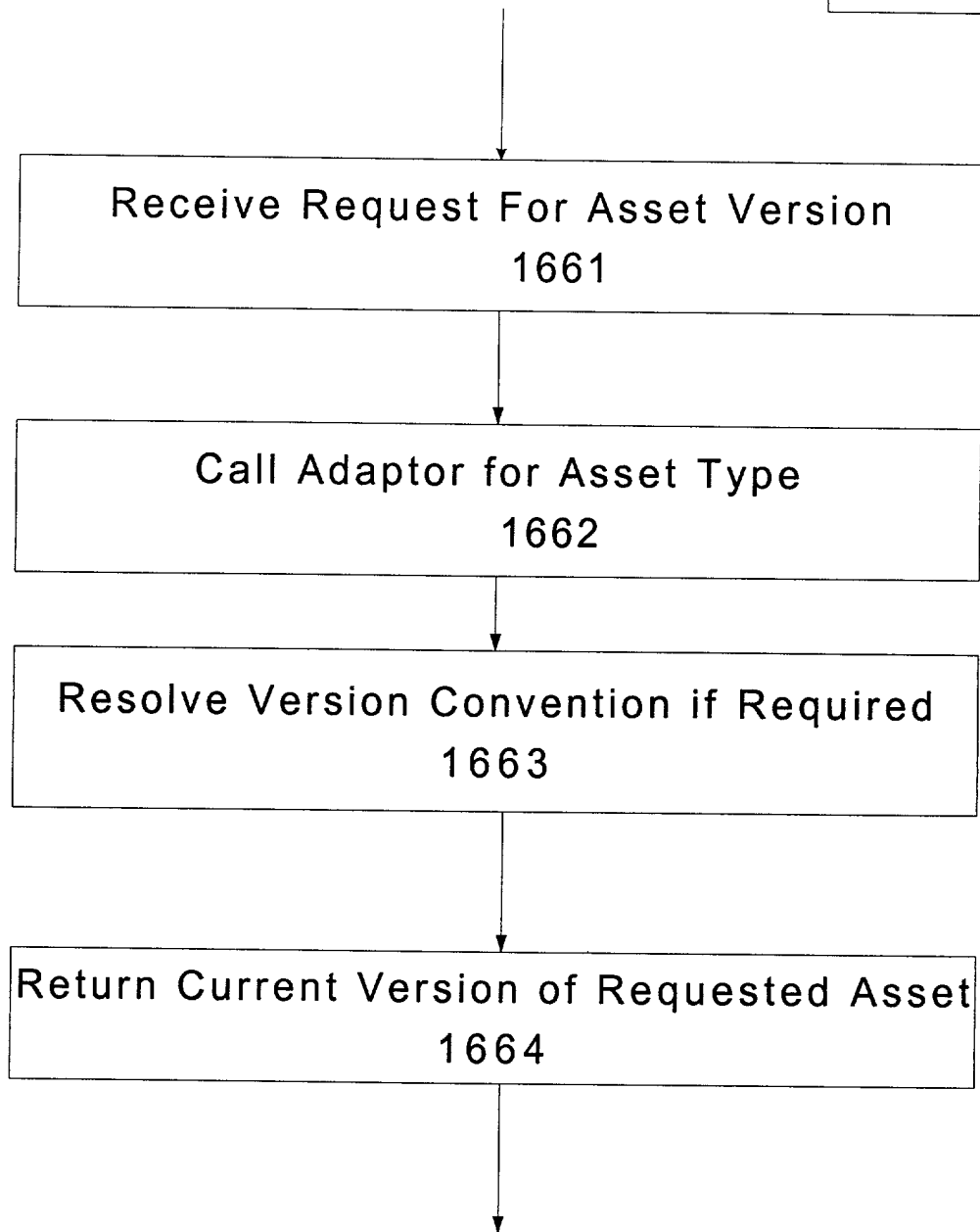
Figure 15A

1600

Export Adapter  
Method

Figure 16

1660



Version Asset Adapter Process - VAM

Figure 16A

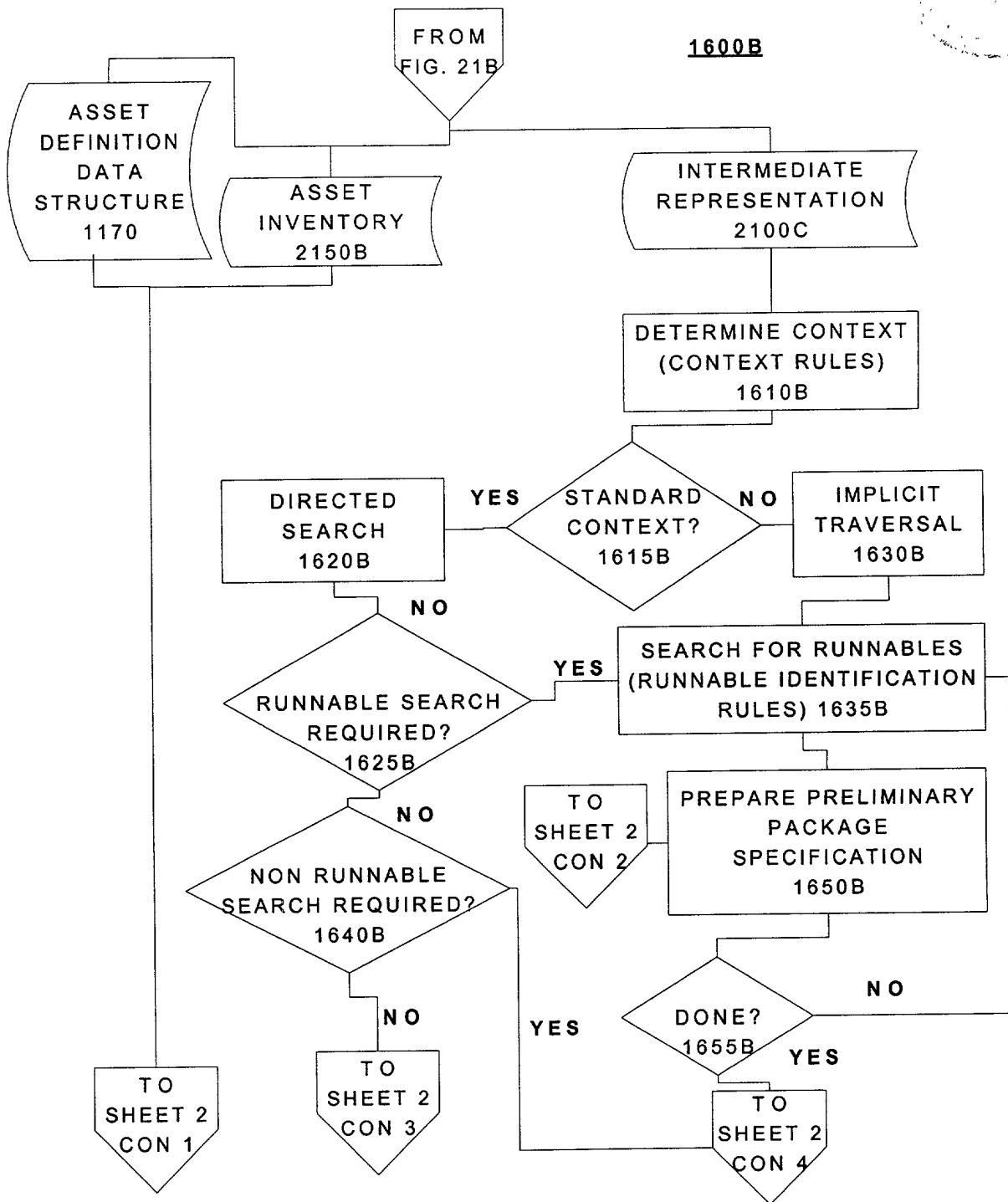


FIG. 16B  
SHEET 1

1600B - 2

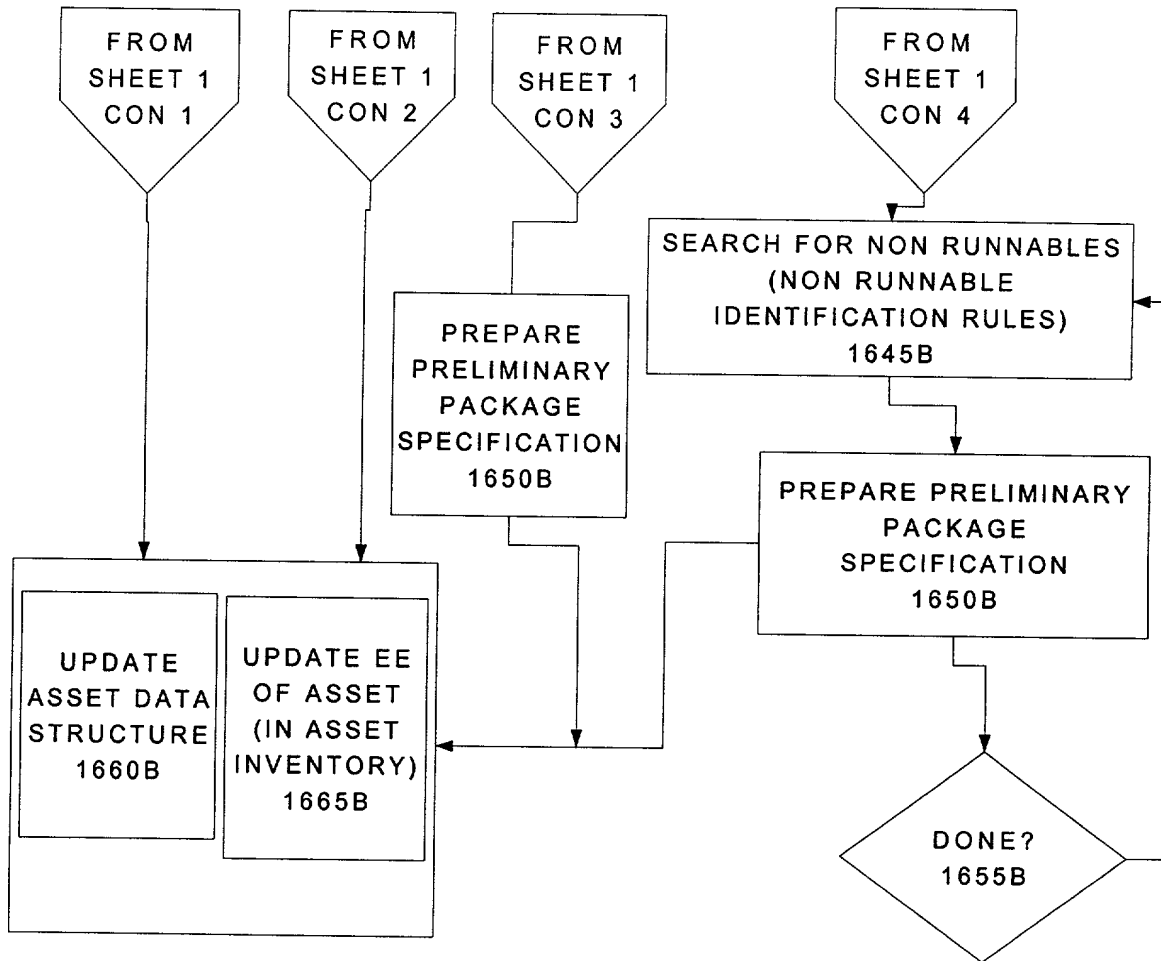
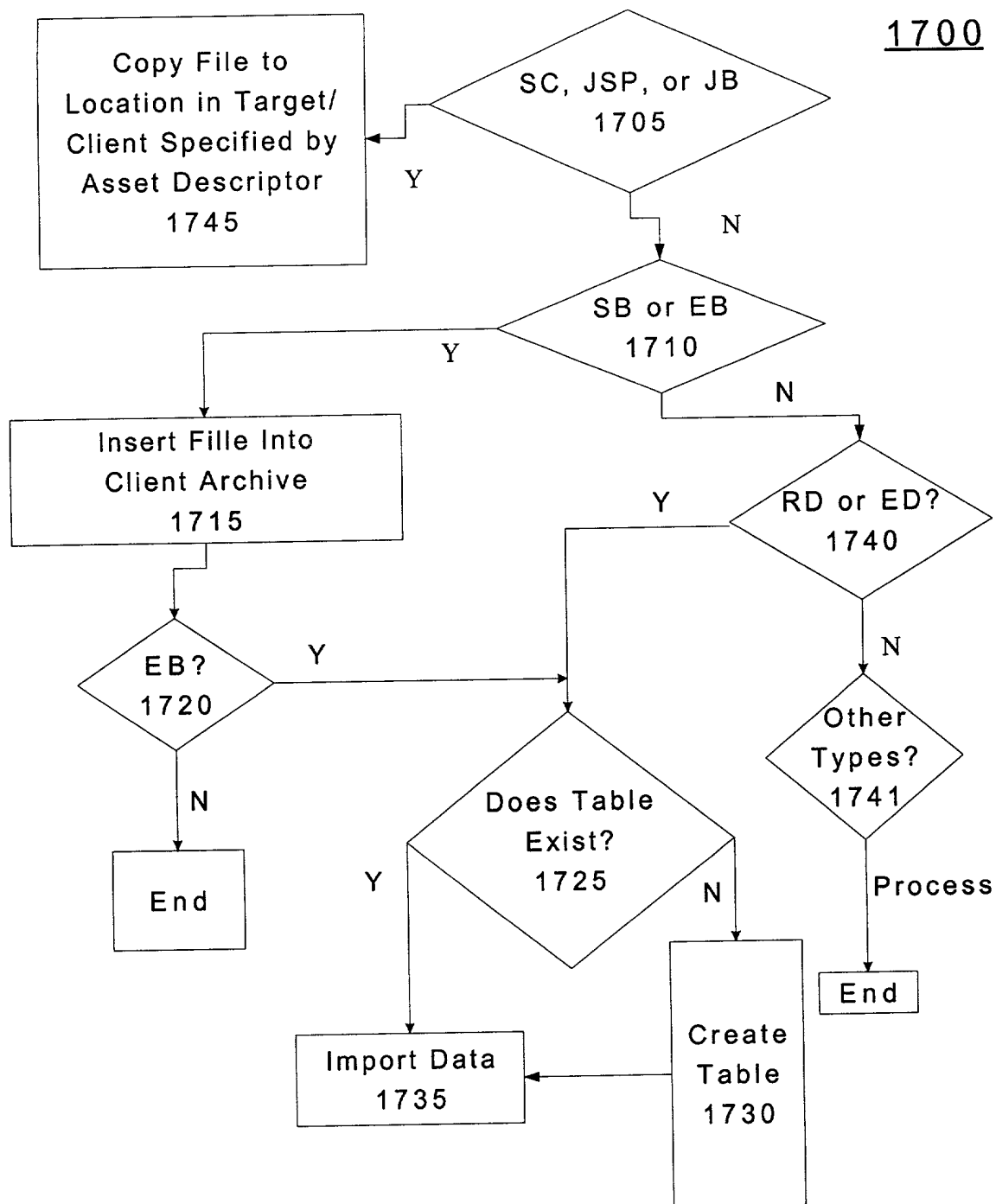


FIG. 16B  
SHEET 2



1700

## Deploy Adapter Method

Figure 17

1700A

DIS TRANSACTIONAL DEPLOYMENT SPHERE OF CONTROL

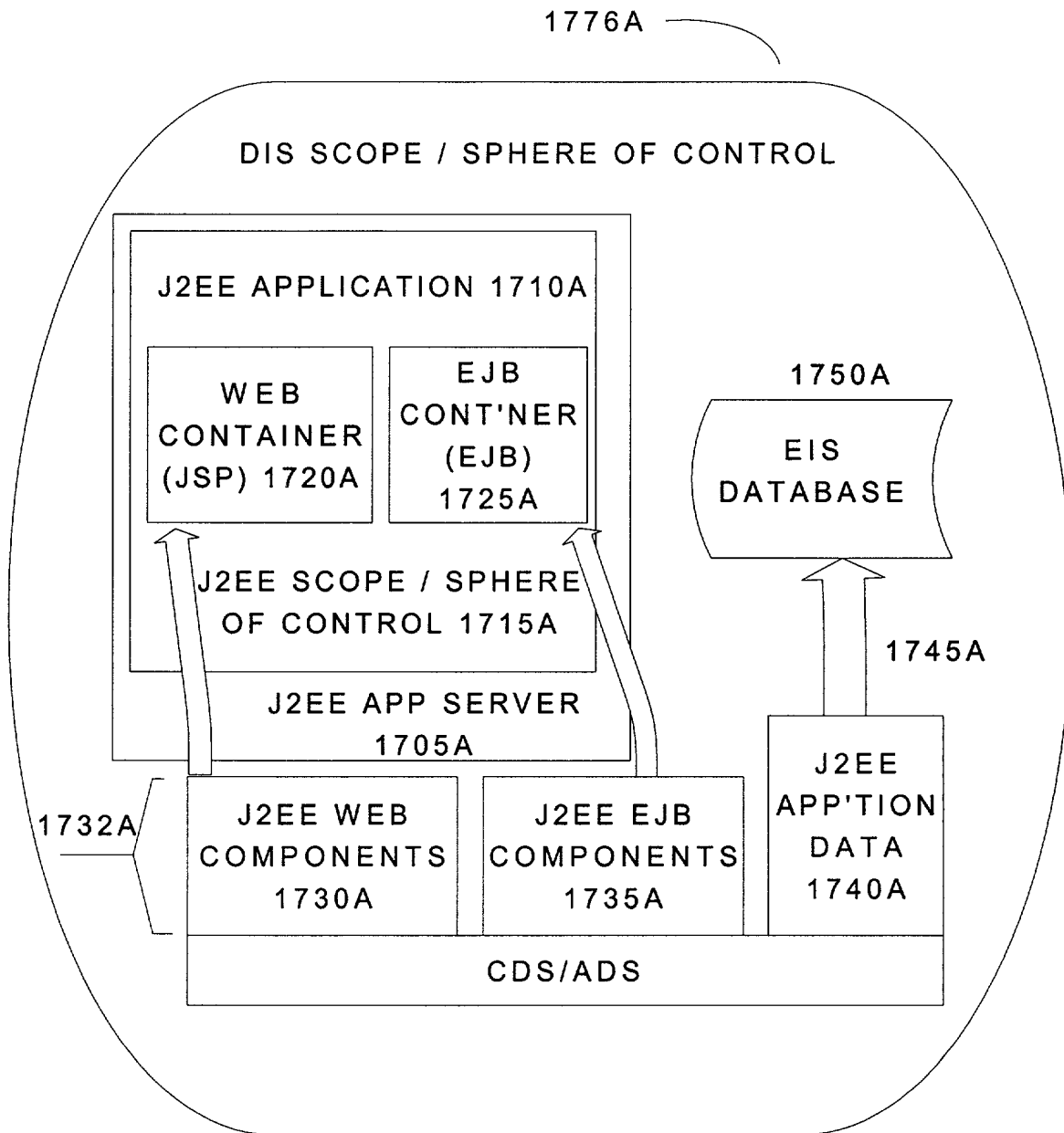


FIGURE 17A

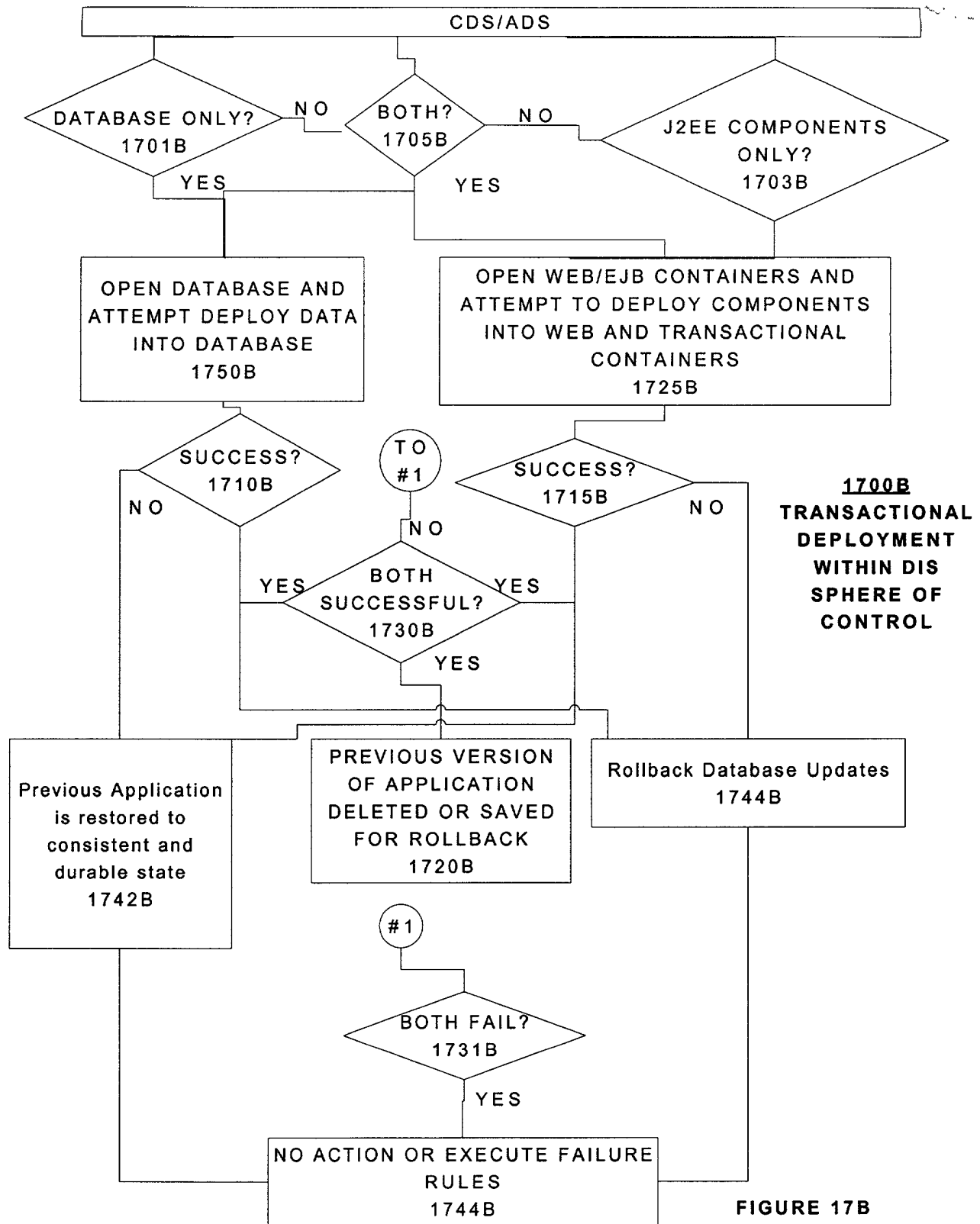
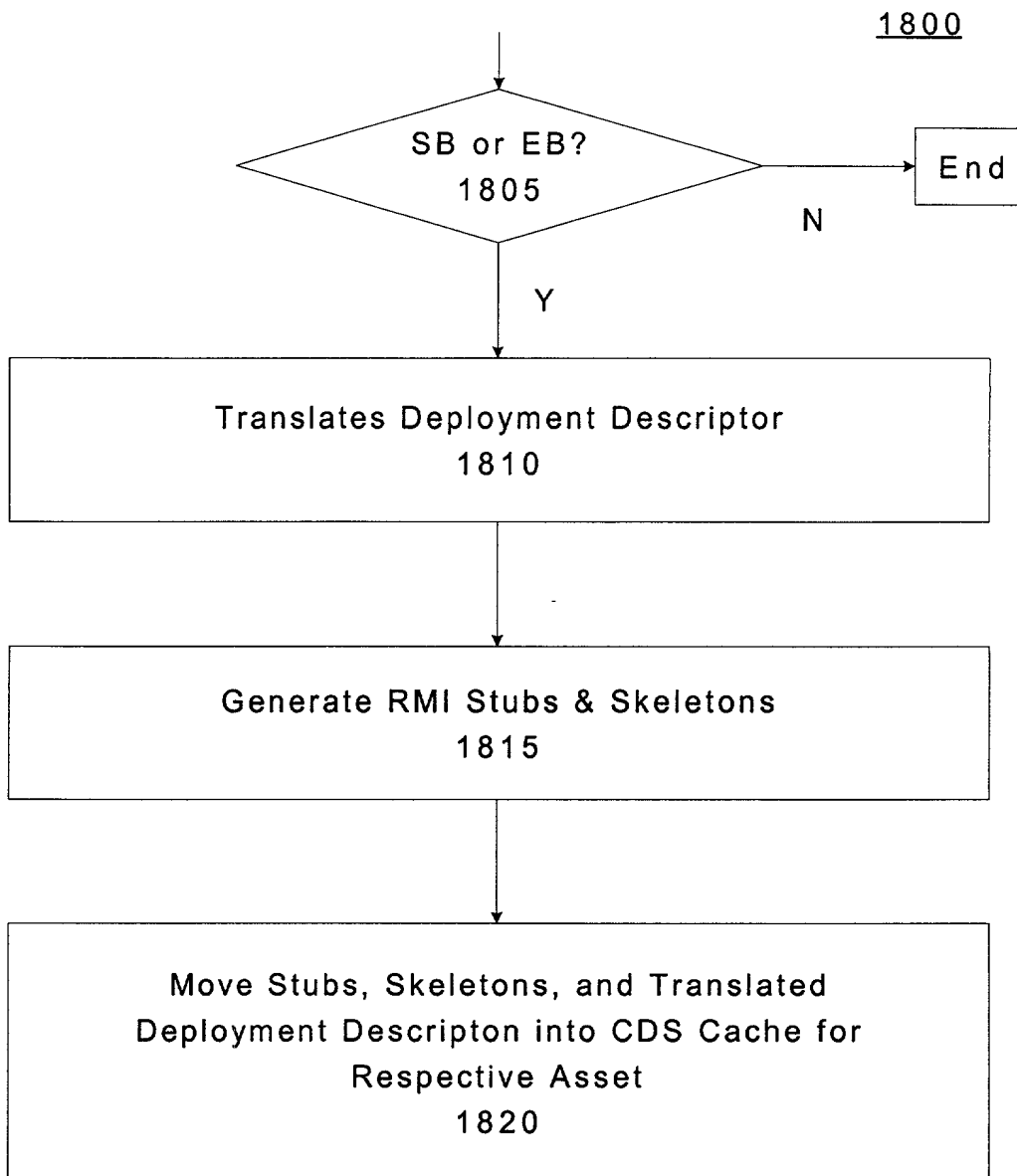
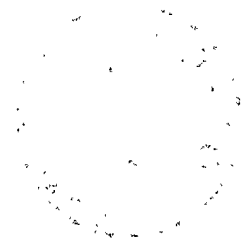


FIGURE 17B



## Process Adaptor Method

Figure 18

1800A

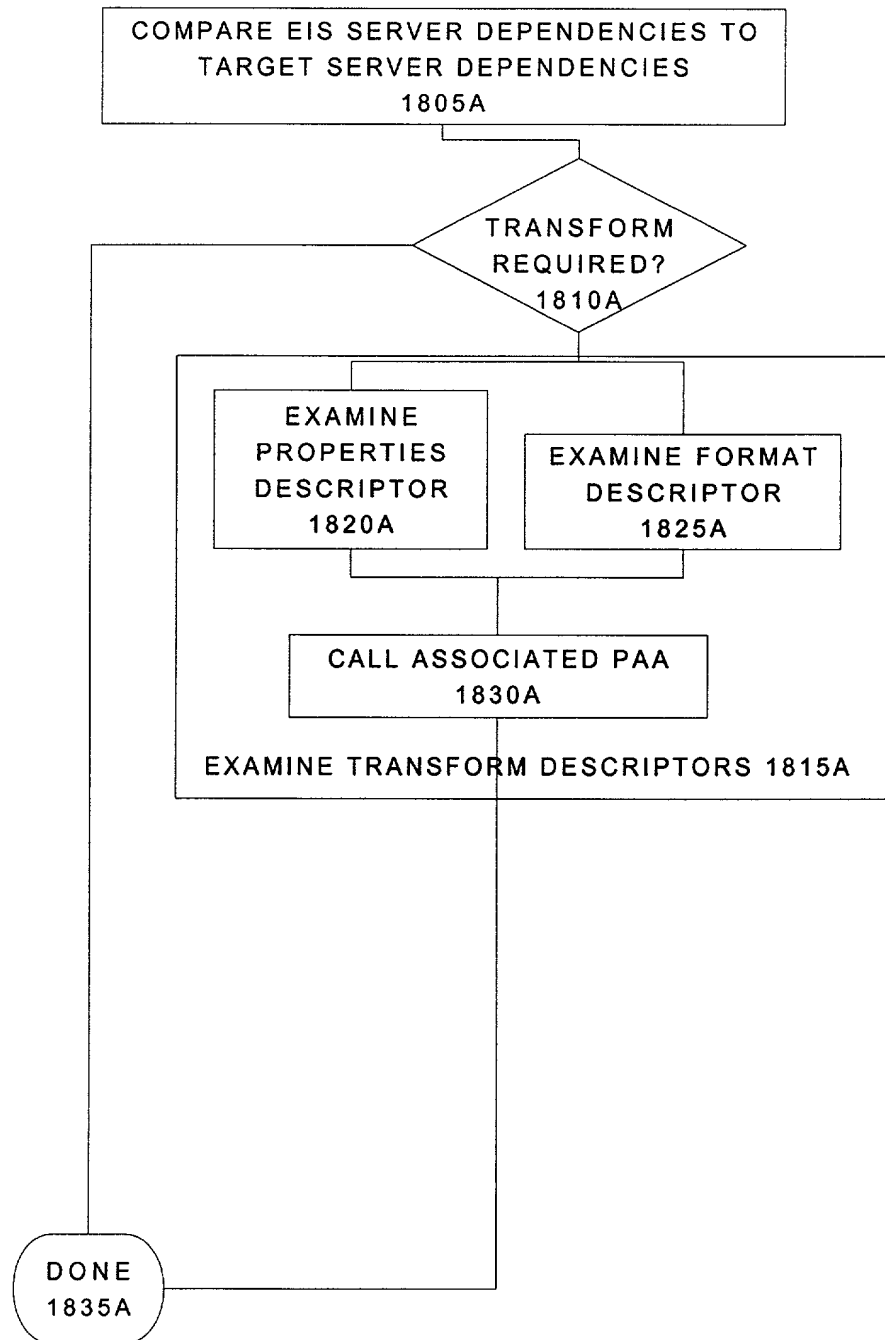
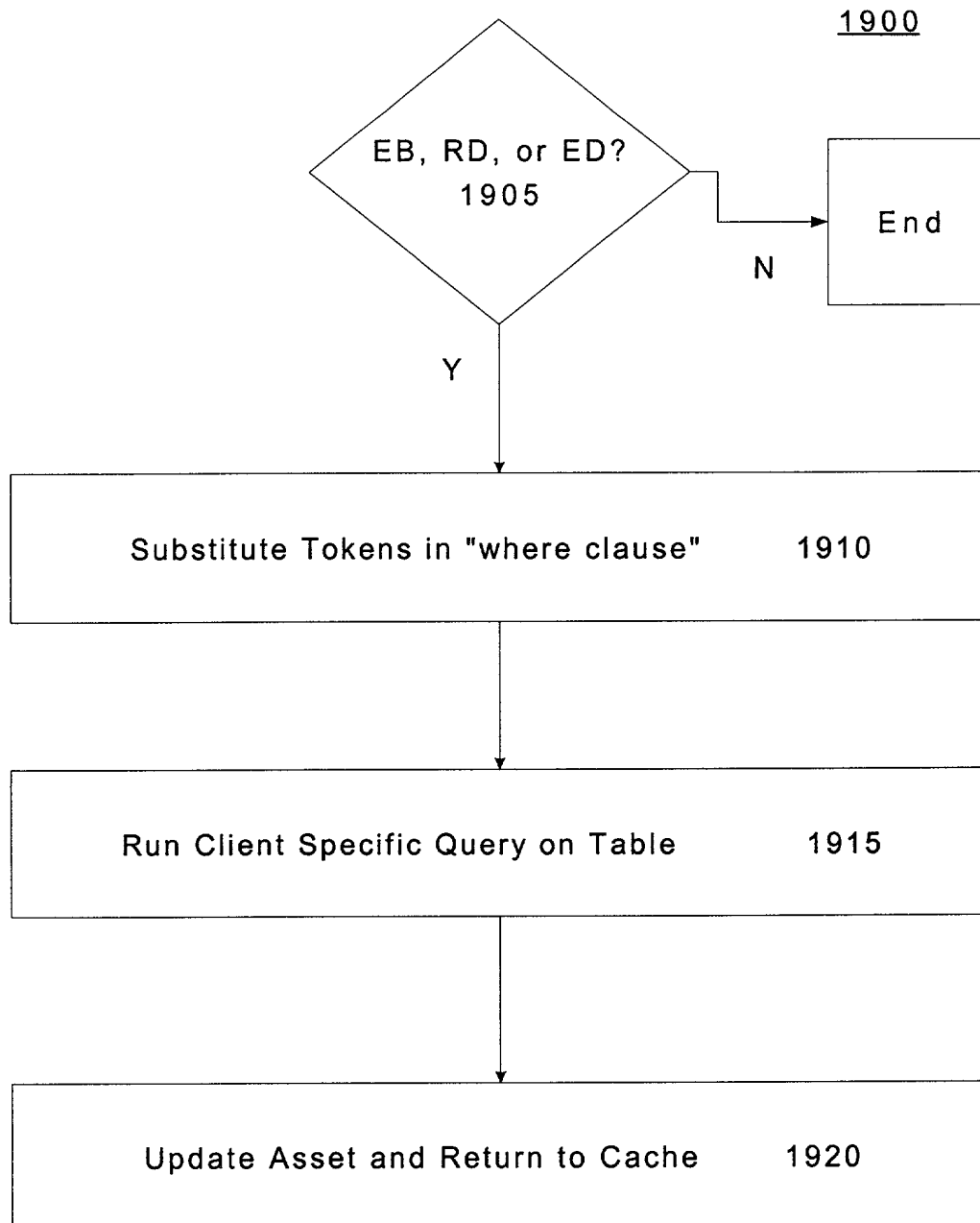
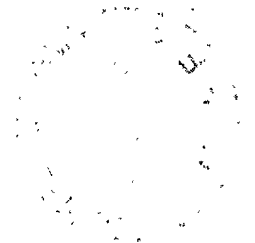
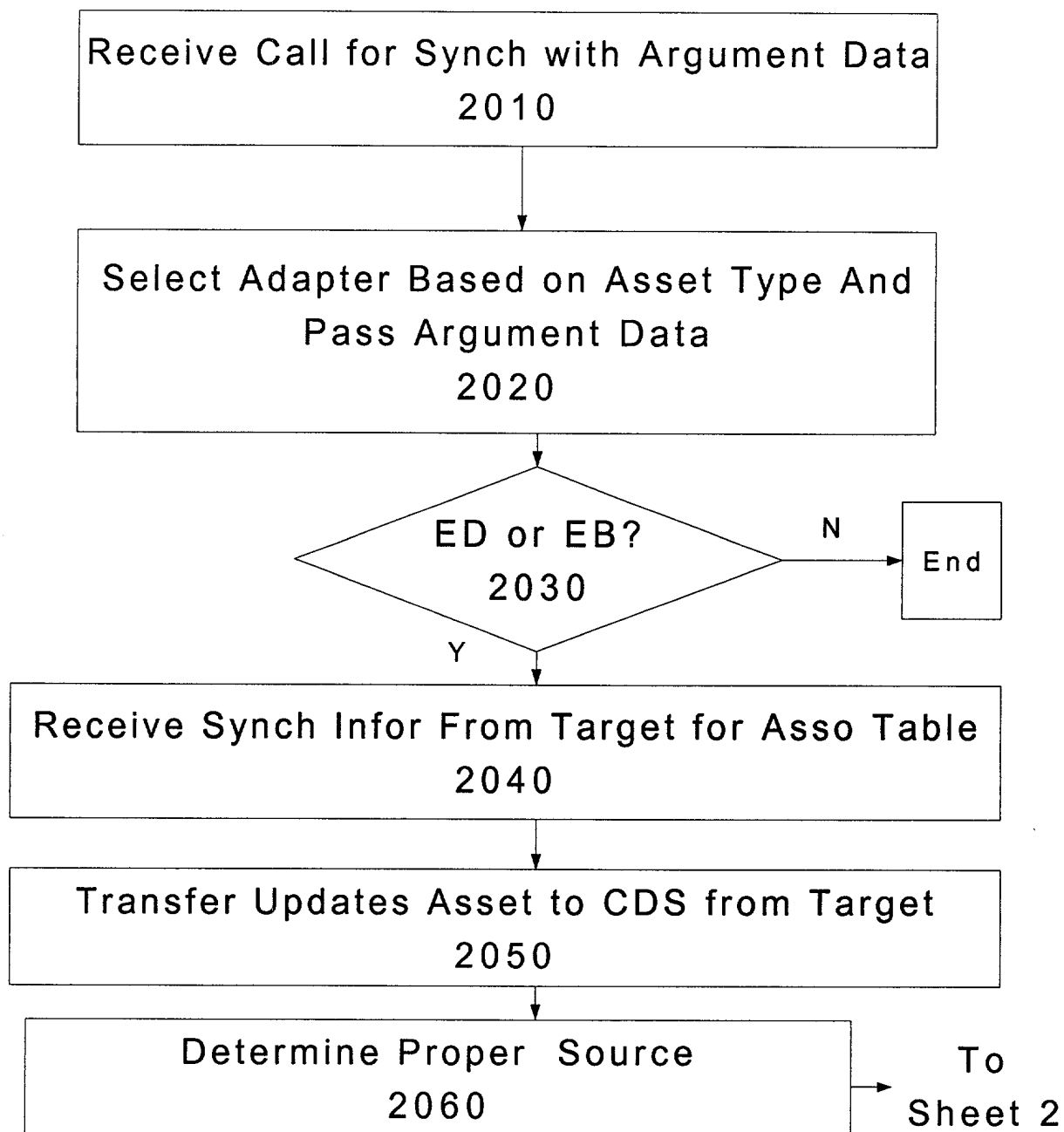


FIGURE 18A



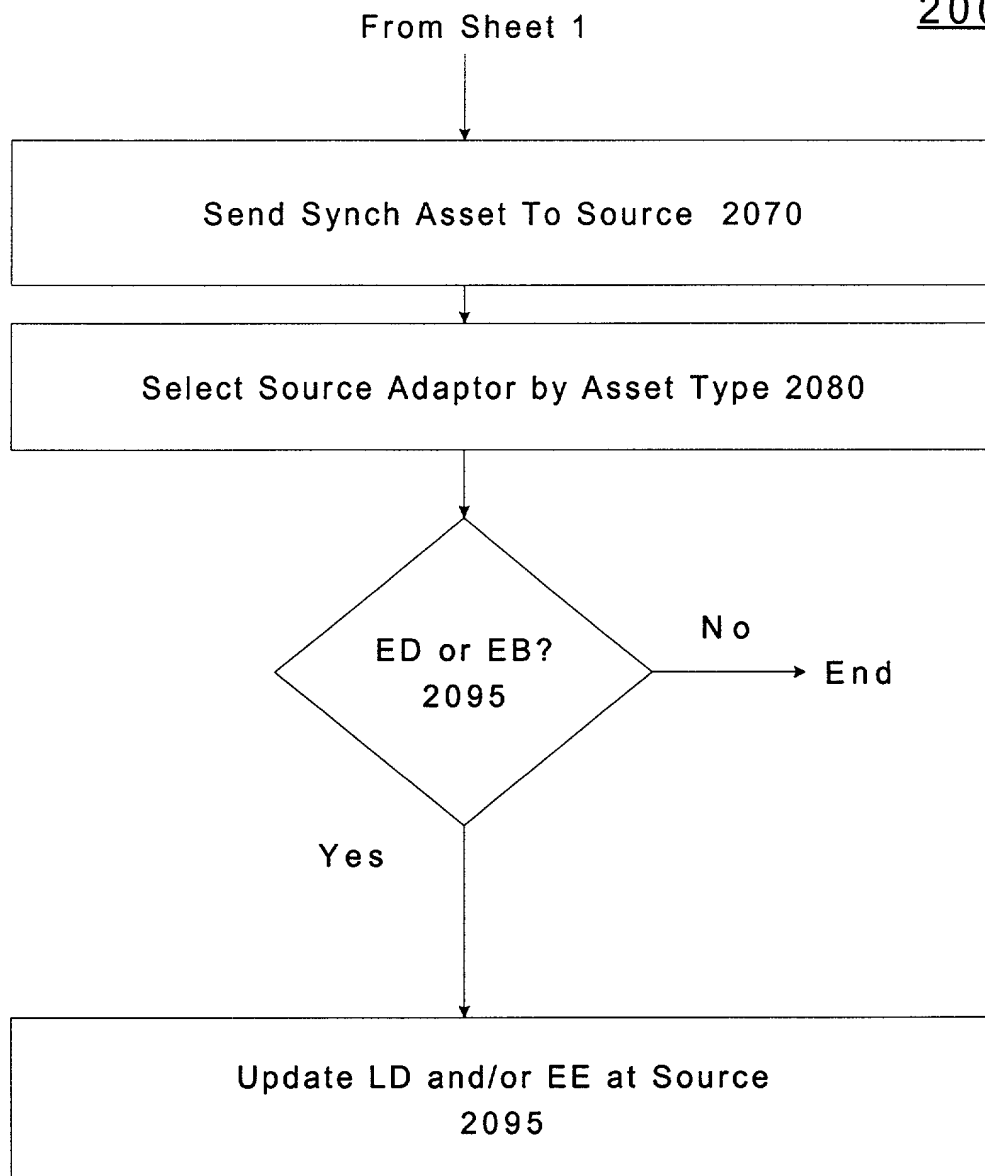
### Target Adapter Method

Figure 19



Synchronize Asset Adapter Process

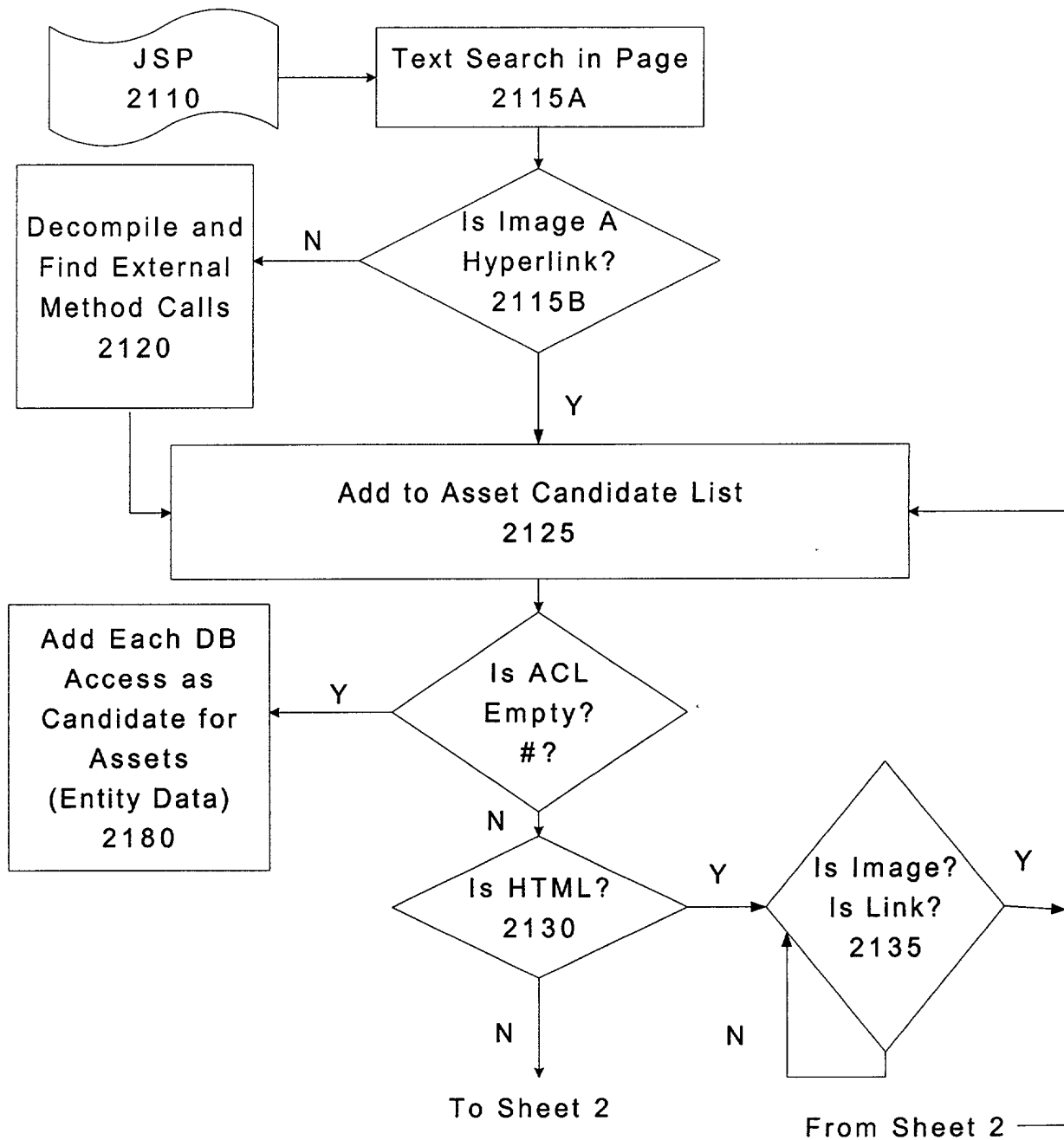
Figure 20 - Sheet 1



Synchronize Asset Adapter Process

Figure 20 - Sheet 2





## Discovery Asset Adapter Method

Figure 21 - Sheet 1

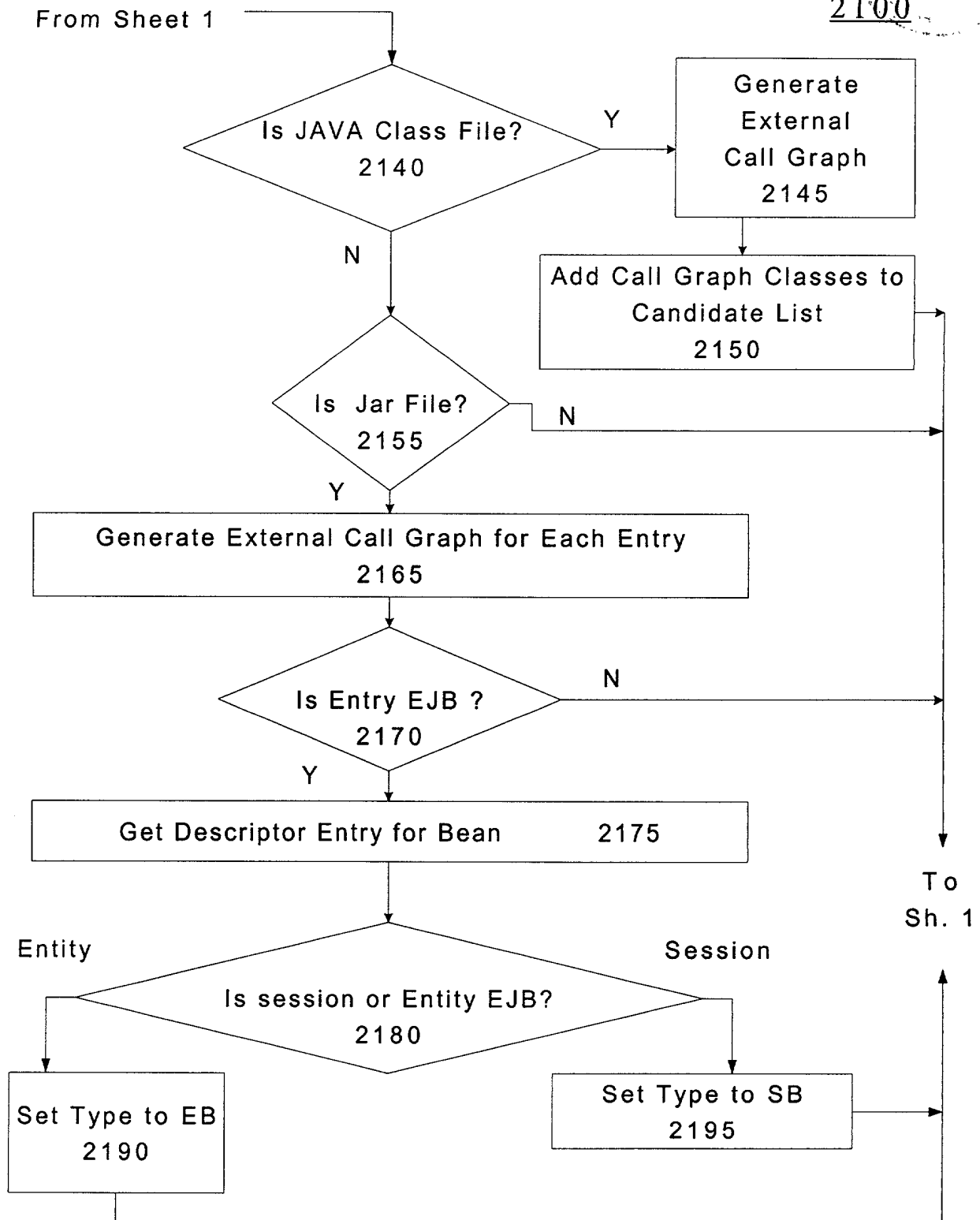
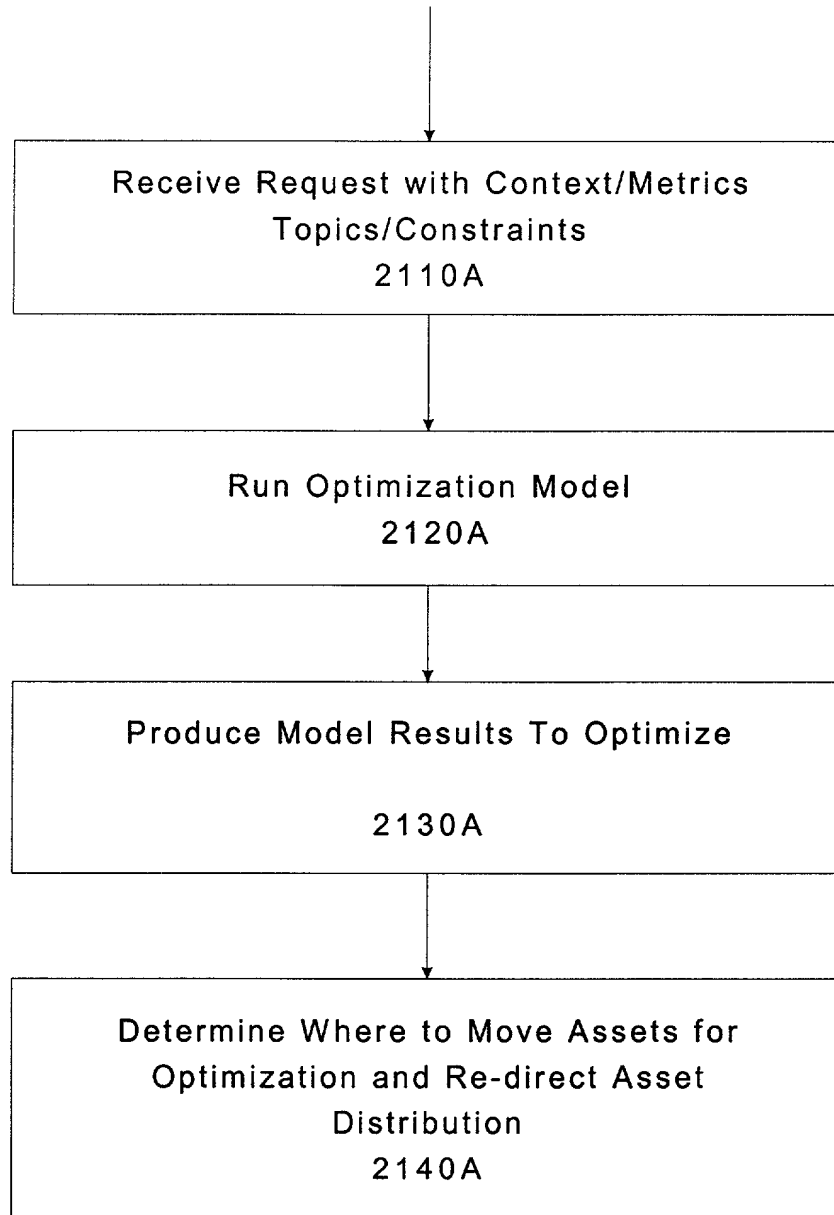


Figure 21-Sheet 2

2100A



Adjustment Asset Adapter Process

Figure 21A

2100B

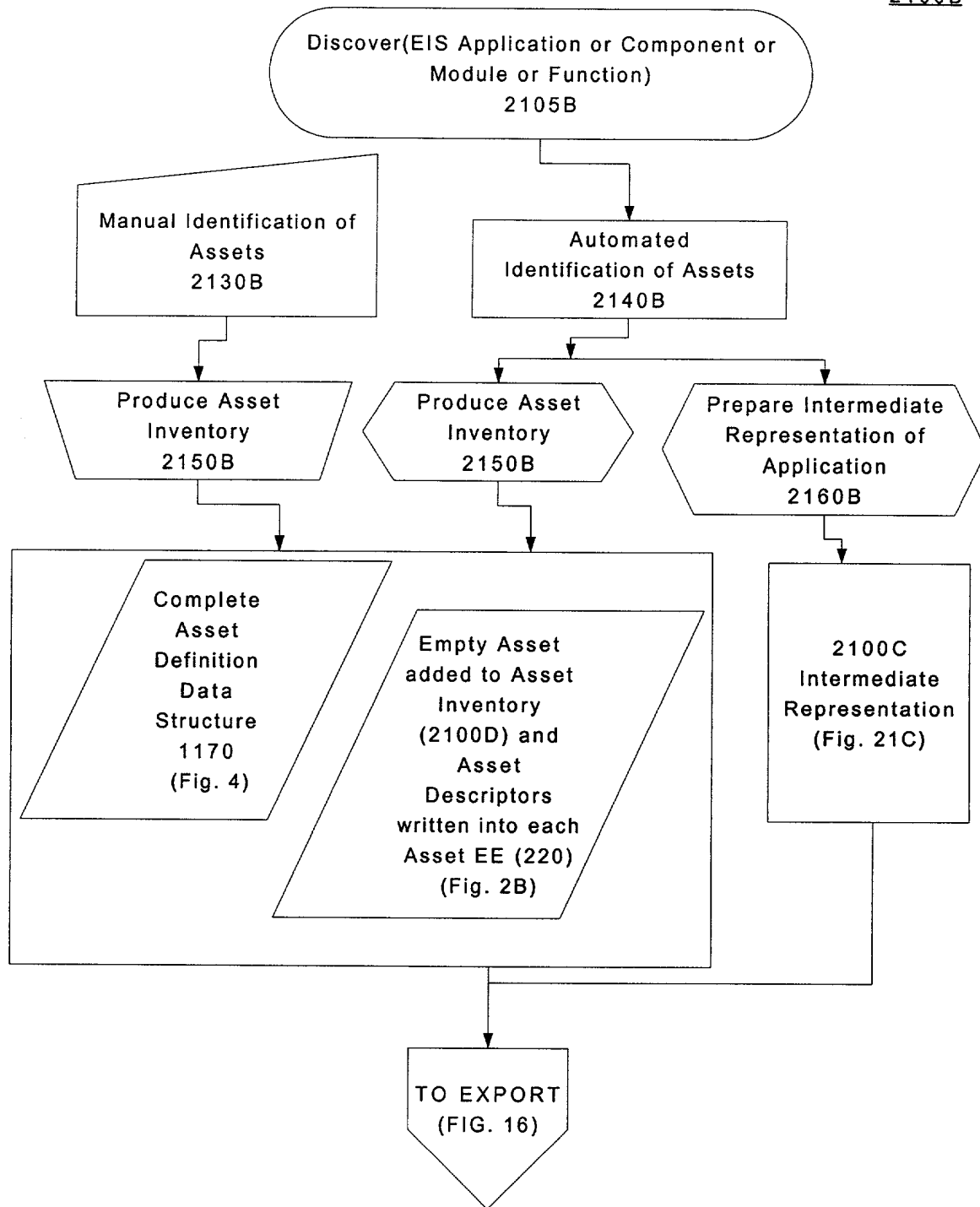
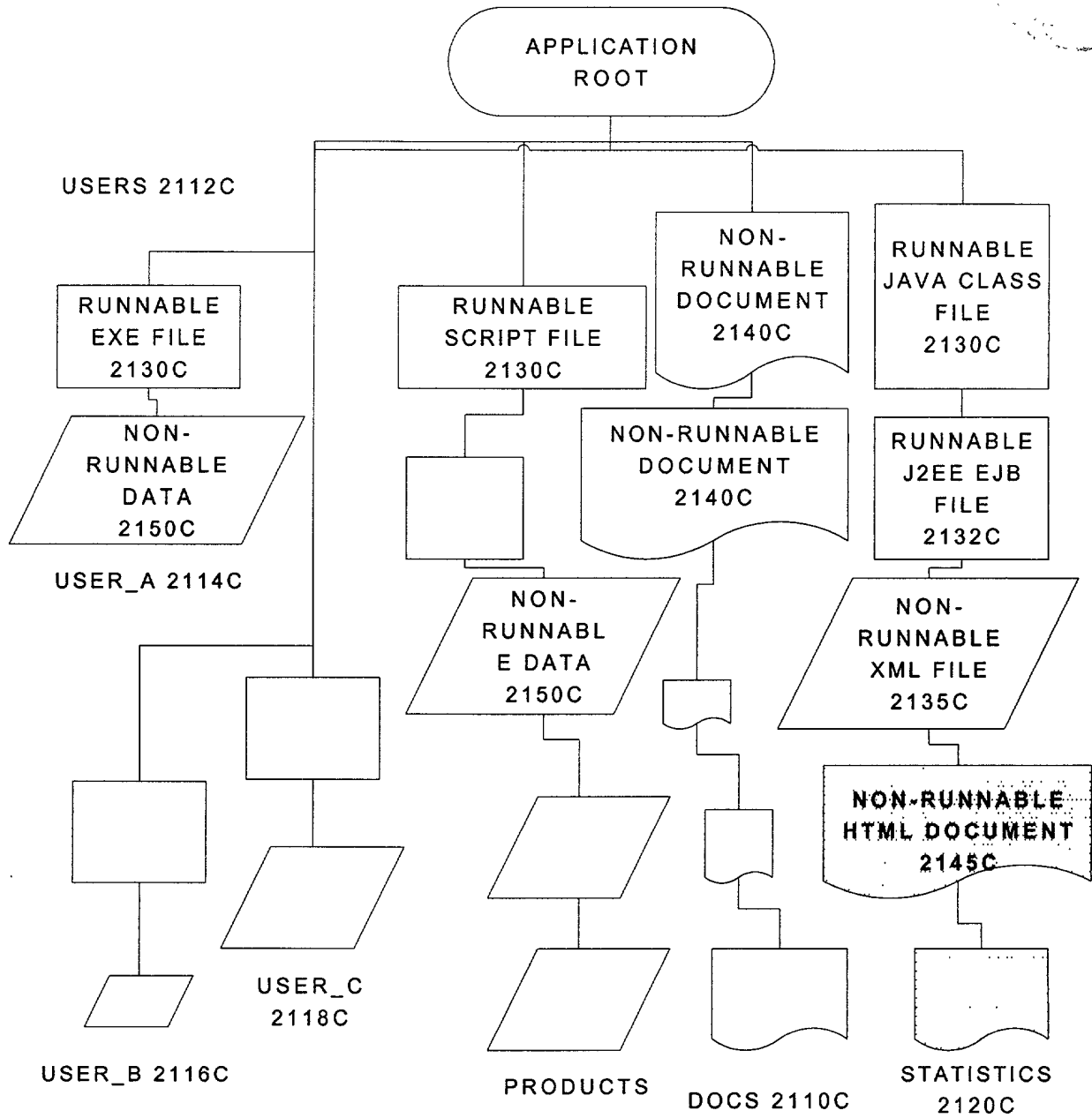


FIGURE 21B

2100C



PRIOR ART

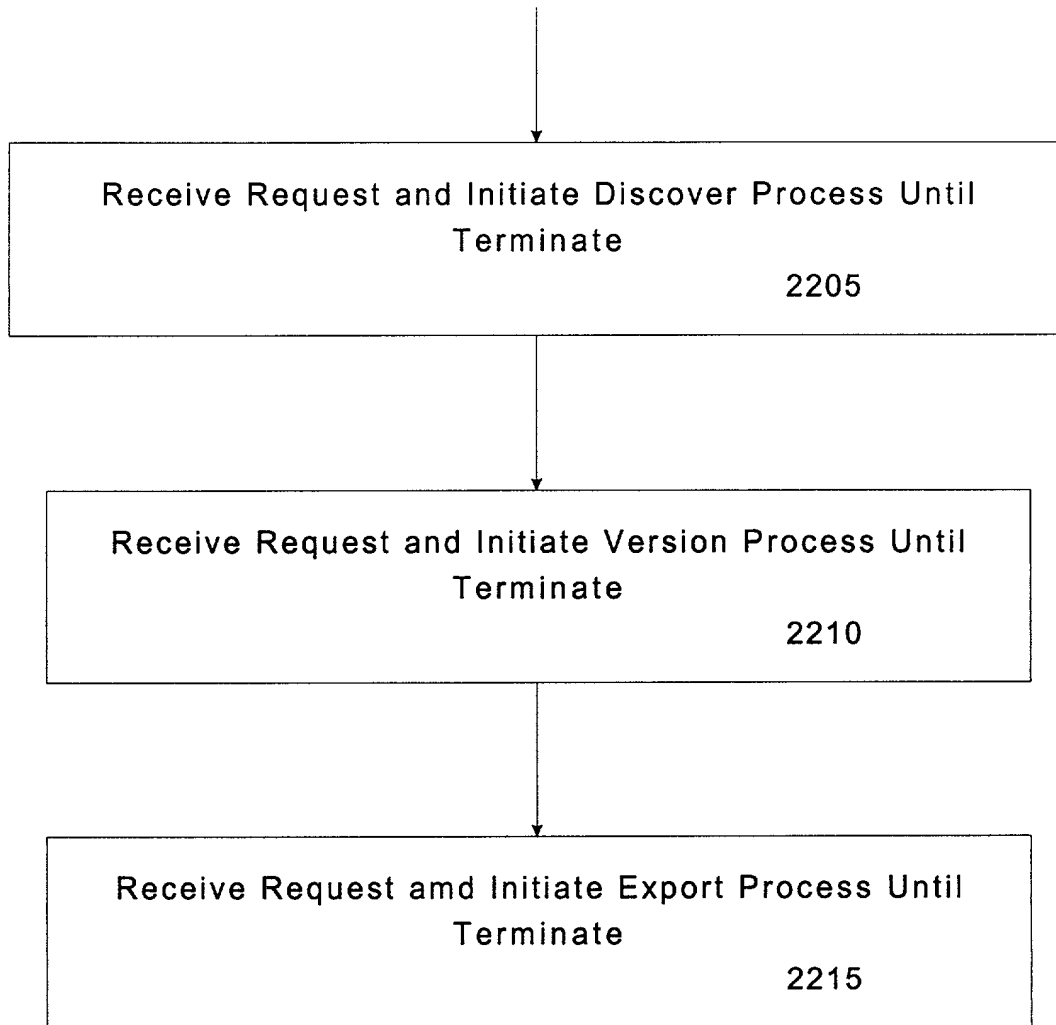
FIGURE 21C

2100D

EE 220	ASSET INTERFACE 230 (OPTIONAL)	LOGIC/DATA (LD) 210
EE 220	ASSET INTERFACE 230 (OPTIONAL)	LOGIC/DATA (LD) 210
EE 220	ASSET INTERFACE 230 (OPTIONAL)	LOGIC/DATA (LD) 210
EE 220	ASSET INTERFACE 230 (OPTIONAL)	LOGIC/DATA (LD) 210
EE 220	ASSET INTERFACE 230 (OPTIONAL)	LOGIC/DATA (LD) 210
EE 220	ASSET INTERFACE 230 (OPTIONAL)	LOGIC/DATA (LD) 210
EE 220	ASSET INTERFACE 230 (OPTIONAL)	LOGIC/DATA (LD) 210



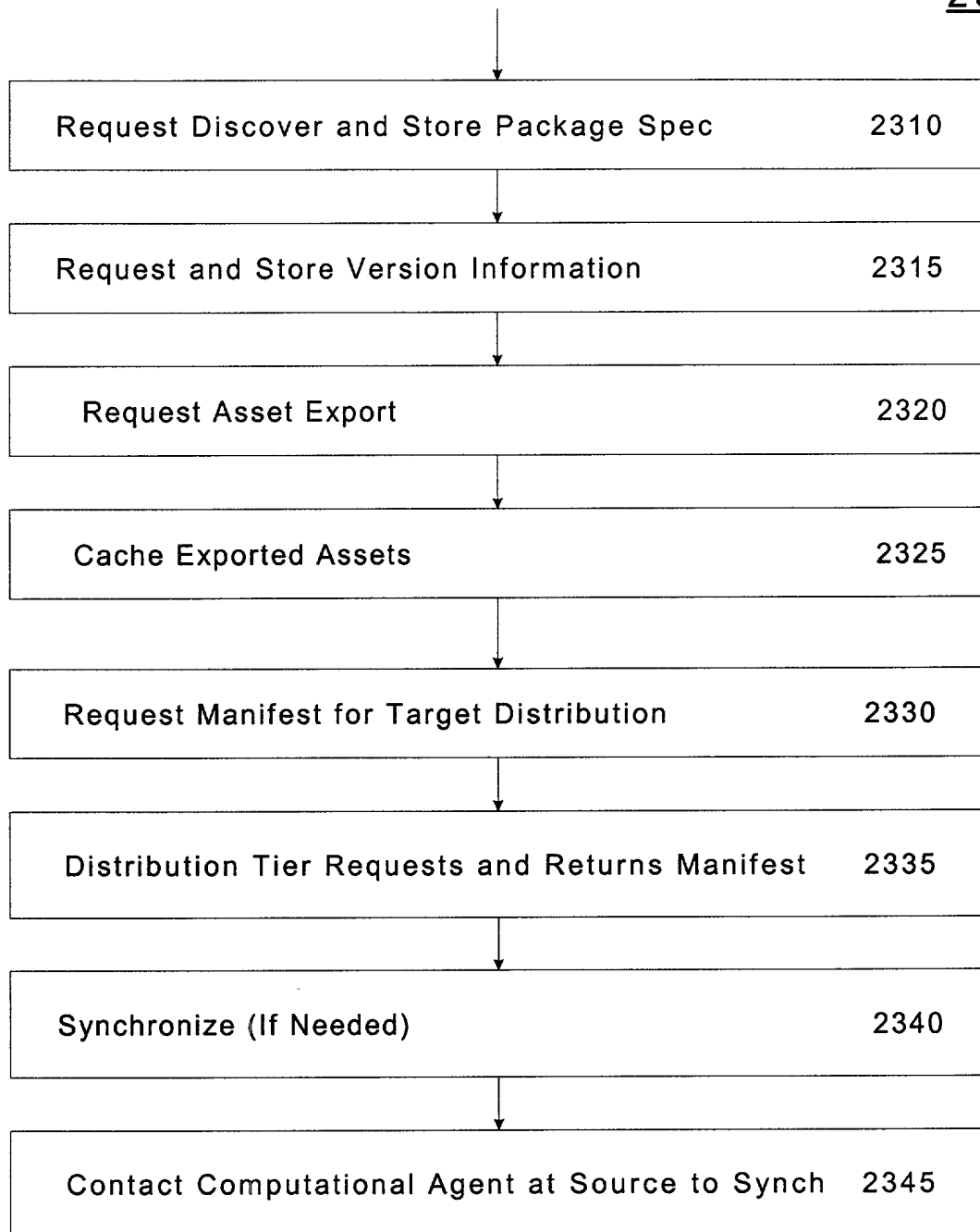
FIGURE 21D



## Publishing Agent Method

Figure 22

2300

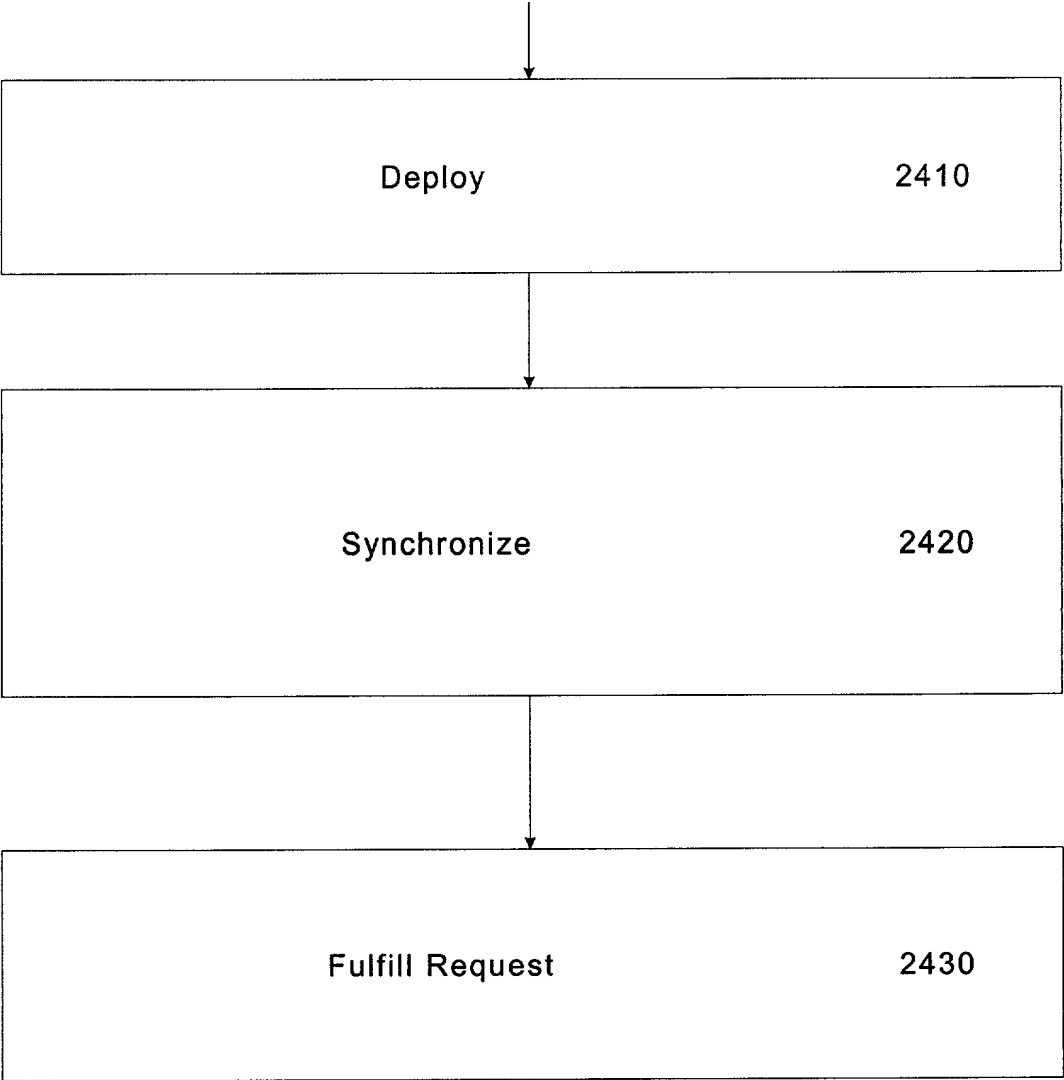


### Subscriber Agent Method

Figure 23



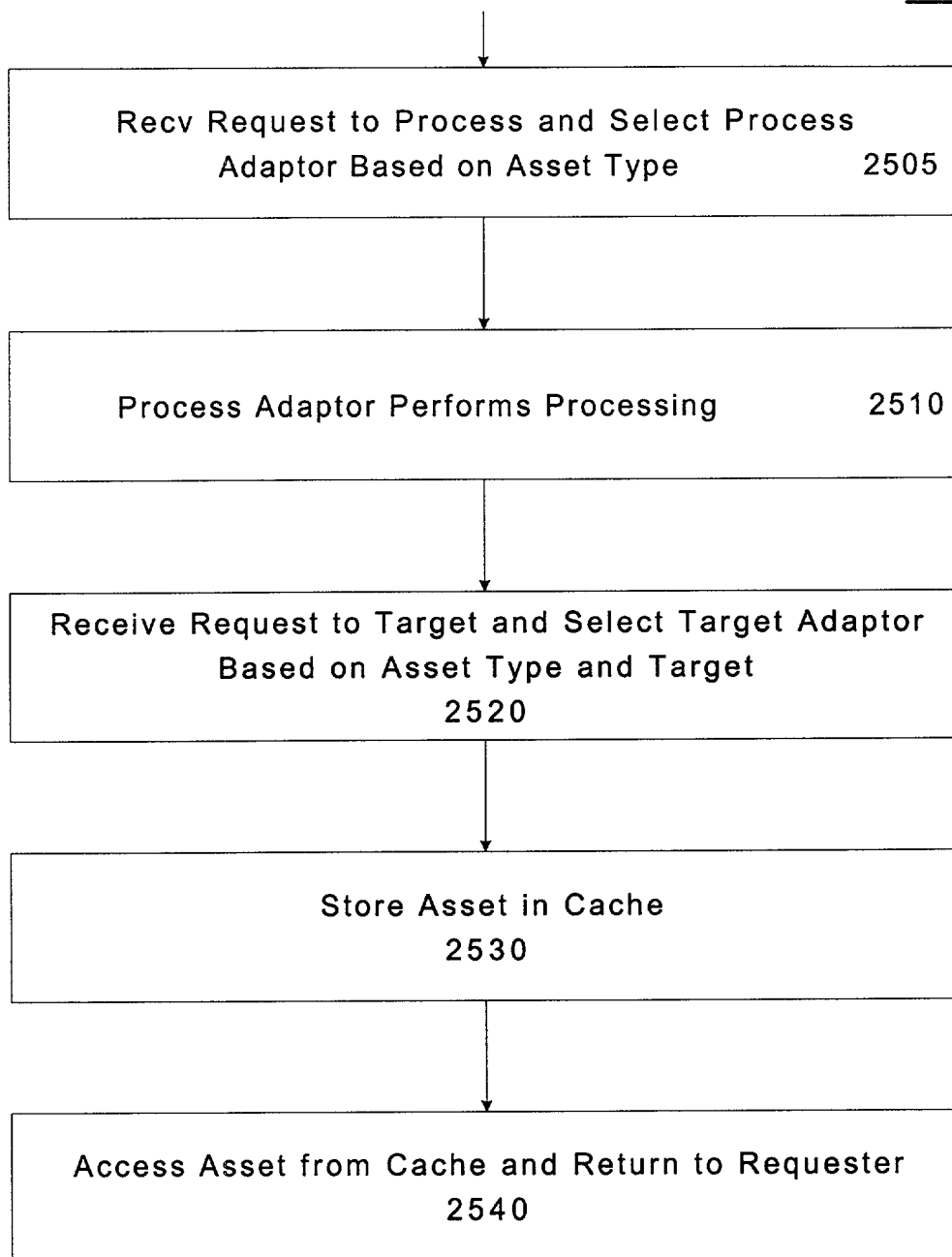
2400



Computational Agent Method

Figure 24

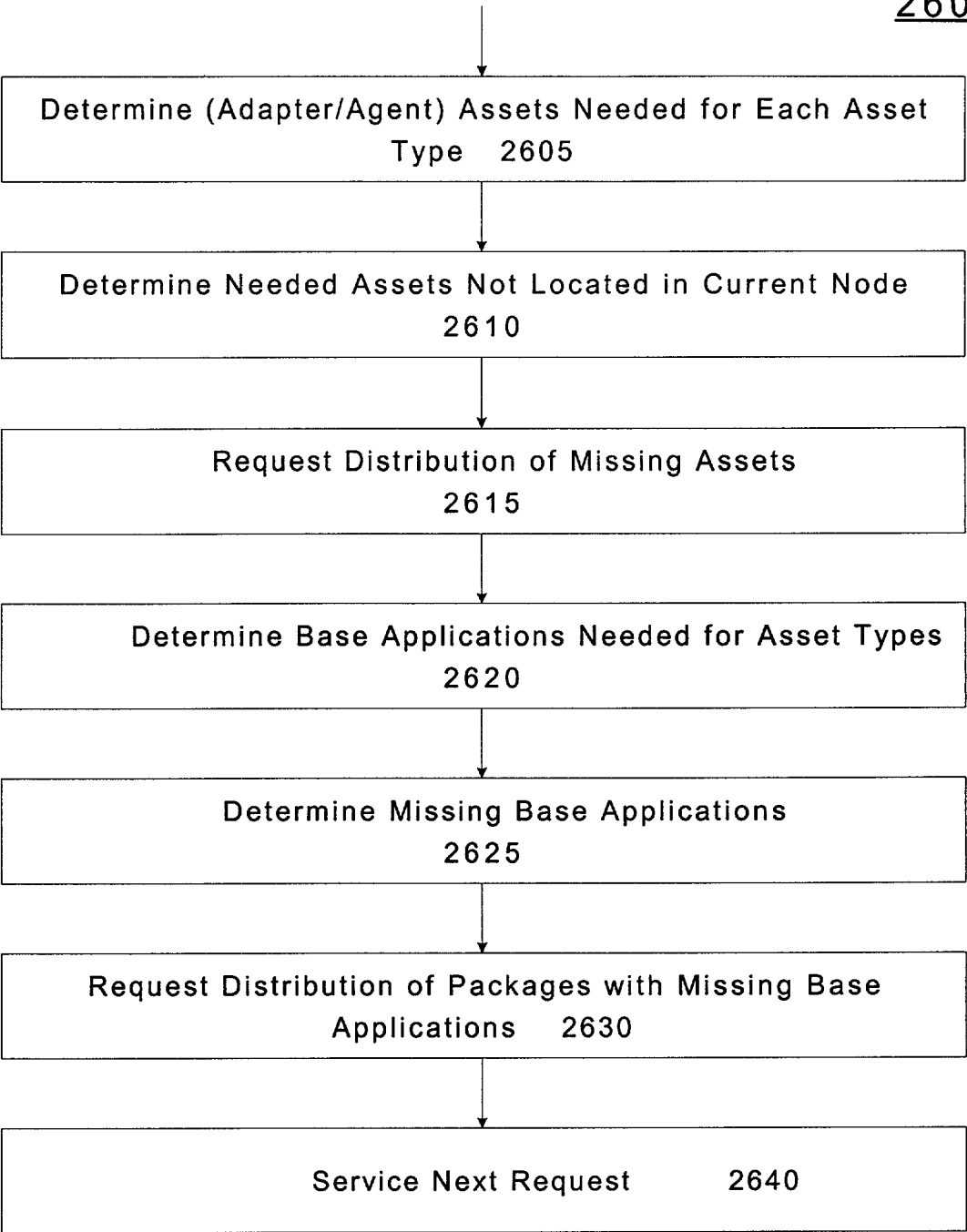
2500



Caching Agent Method

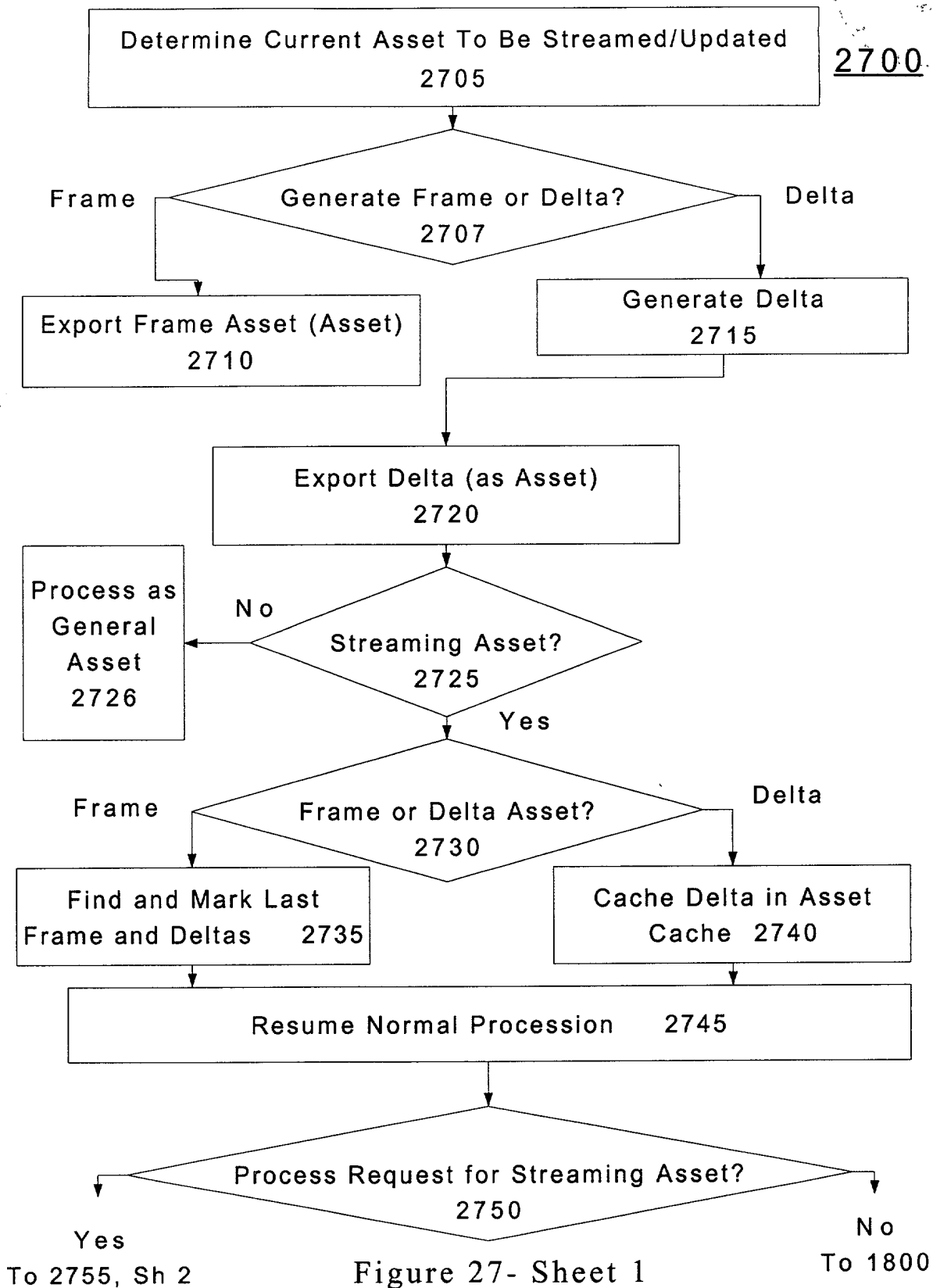
Figure 25

2600



System Asset Distribution Process

Figure 26

2700



From Sheet 1

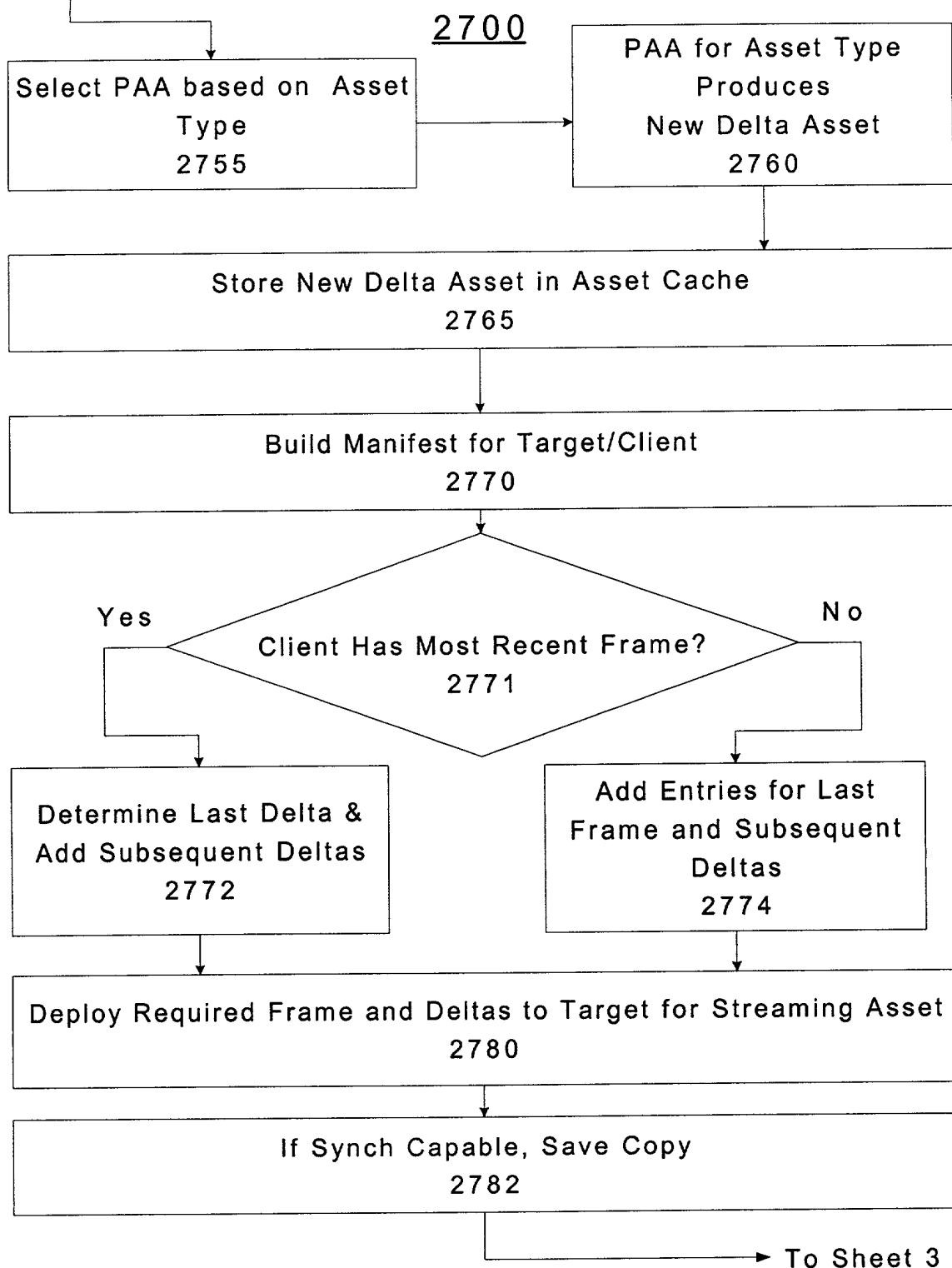


Figure 27-Sheet 2

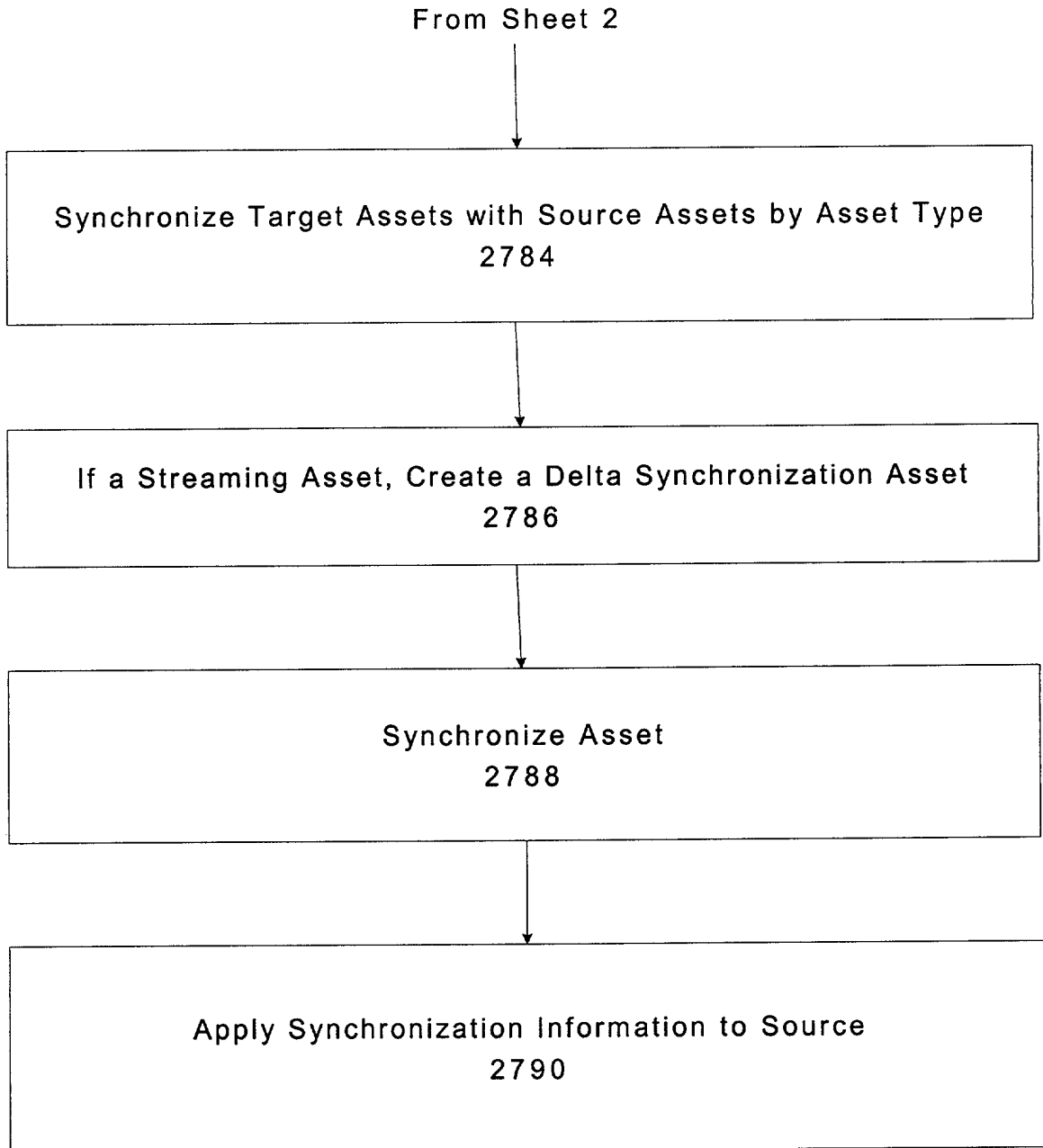
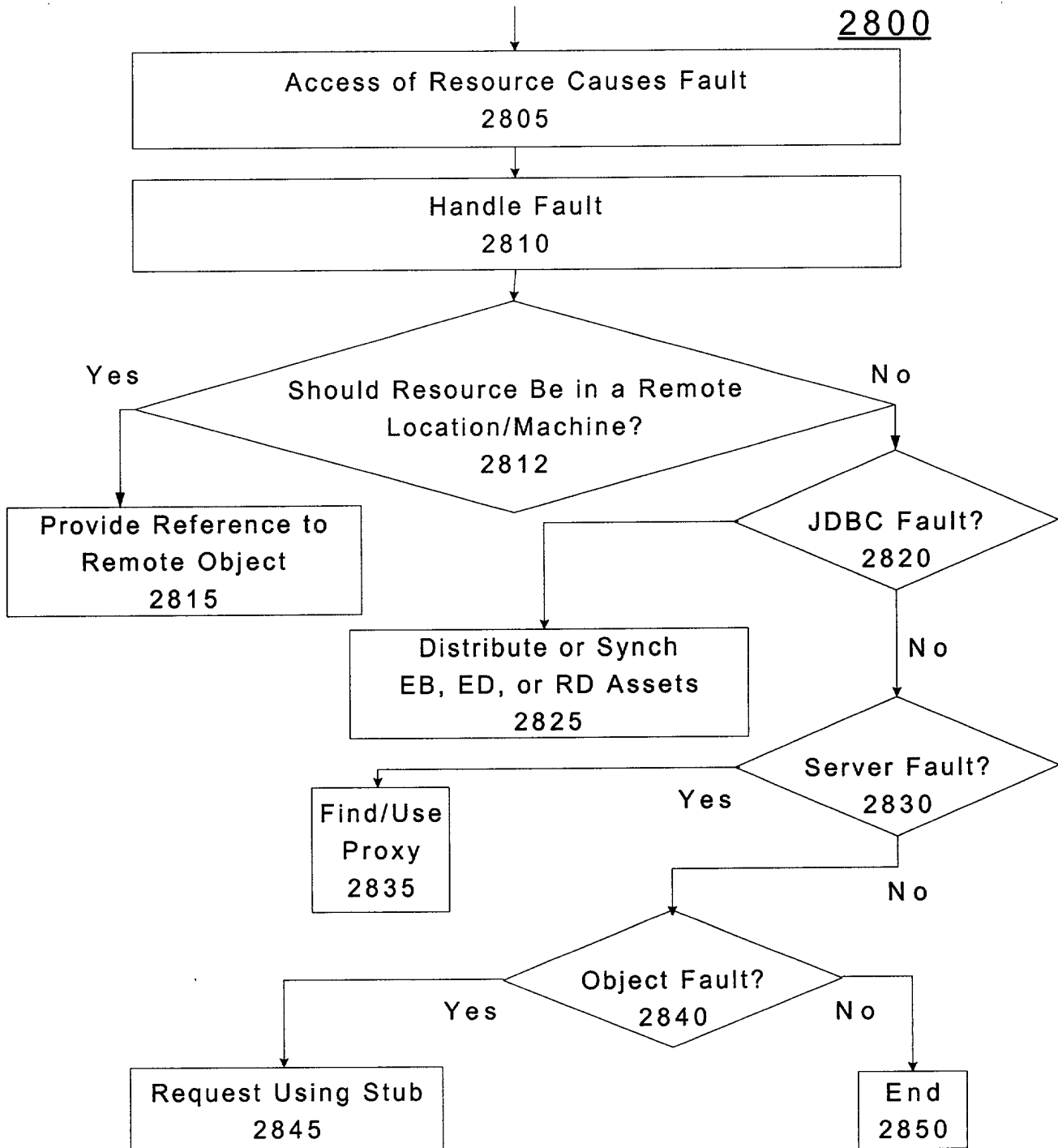


Figure 27-Sheet 3

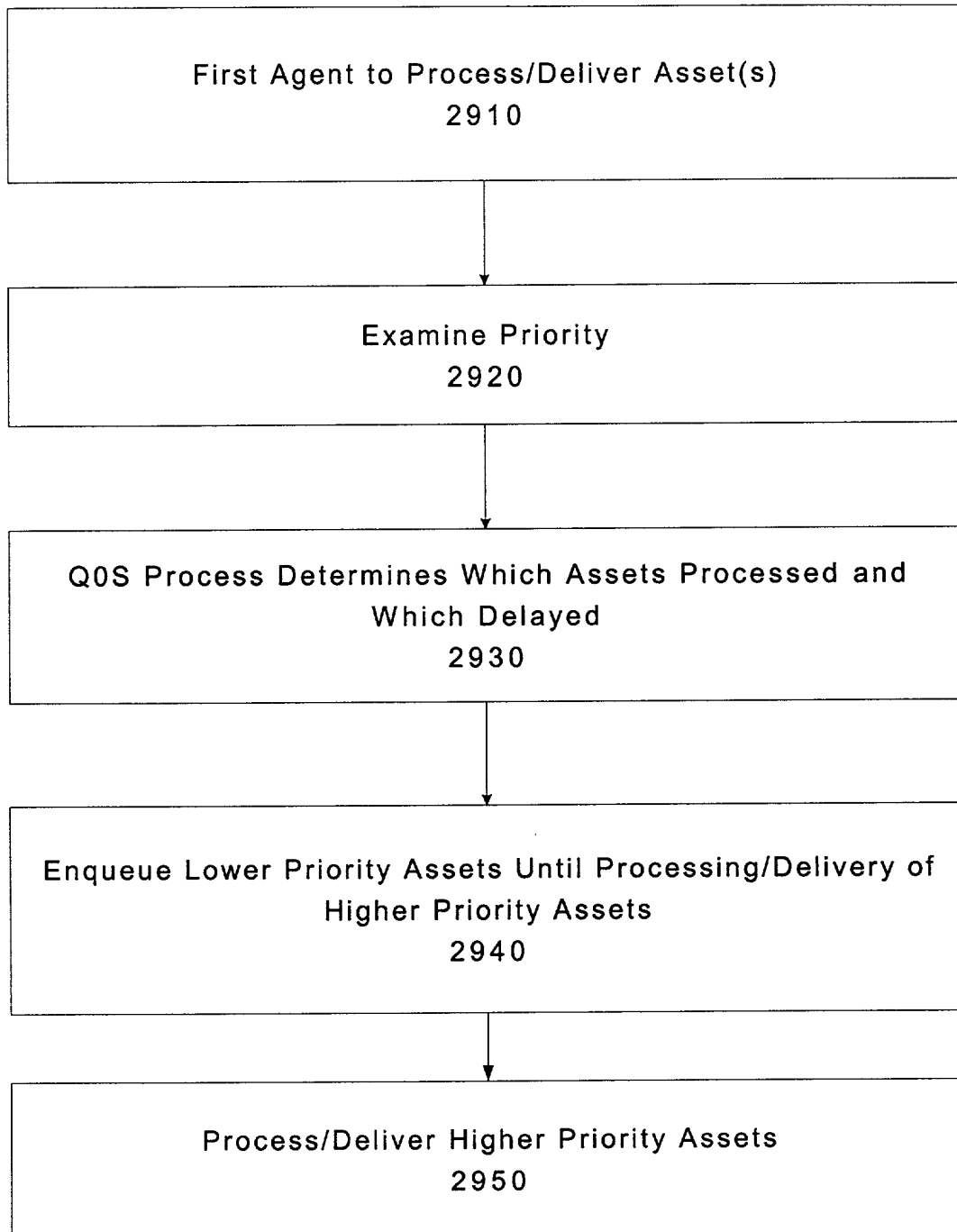
2800



Bridging Process

Figure 28

2900



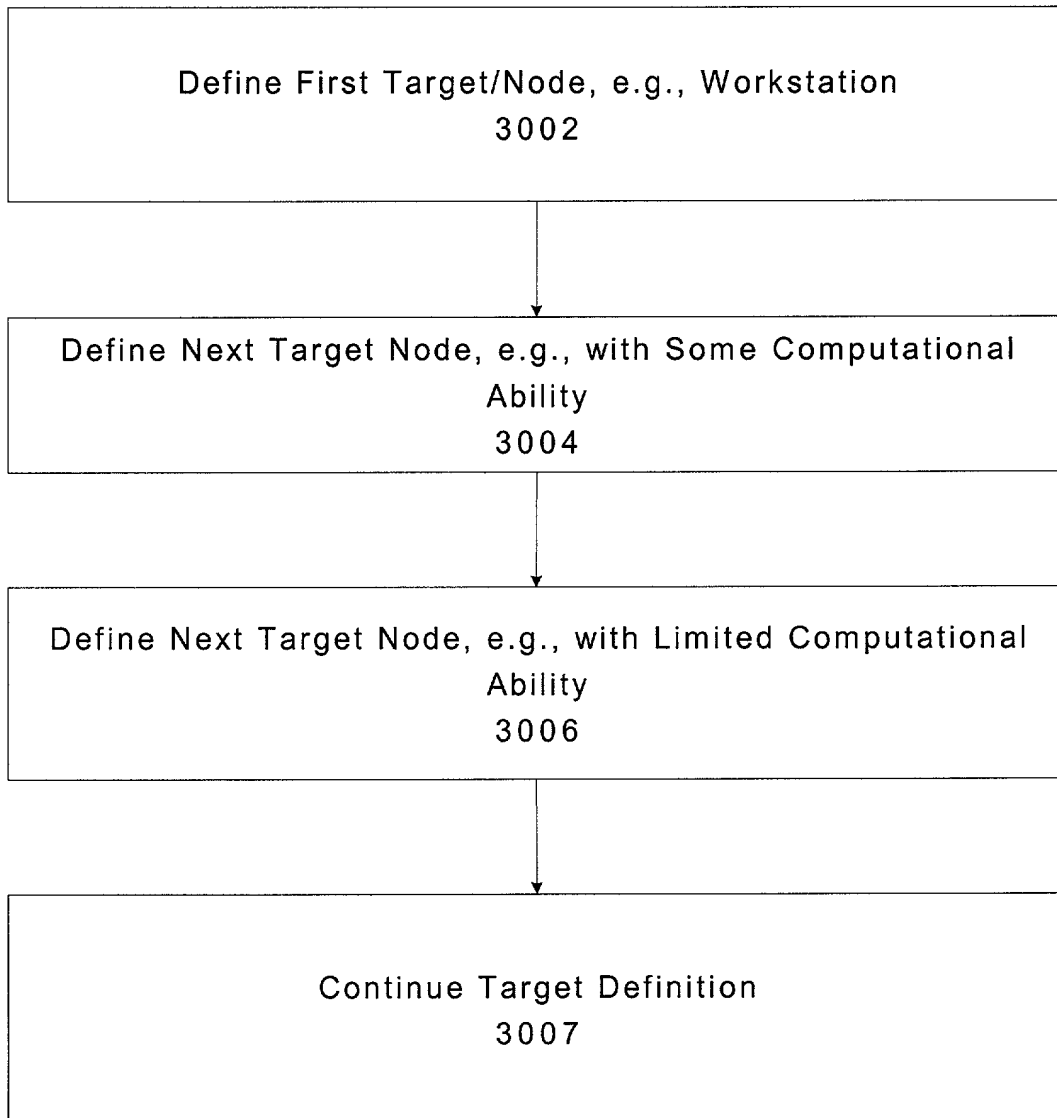
QoS

Figure 29



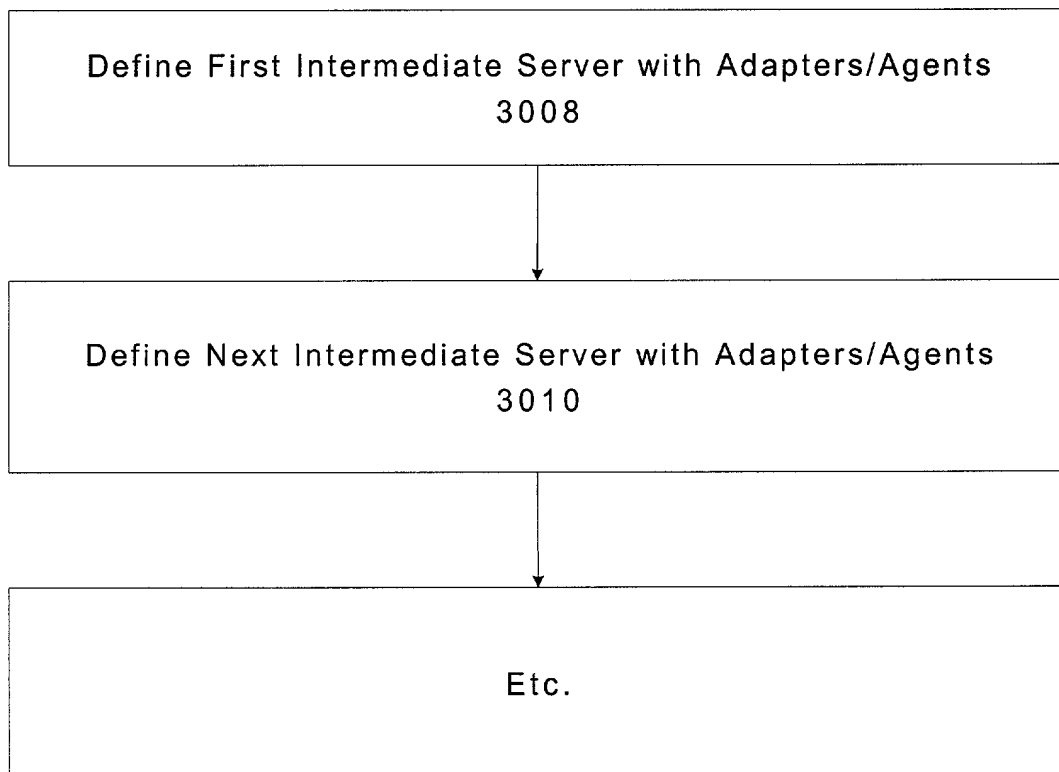


3000A



Target/Client Definition  
Figure 30A

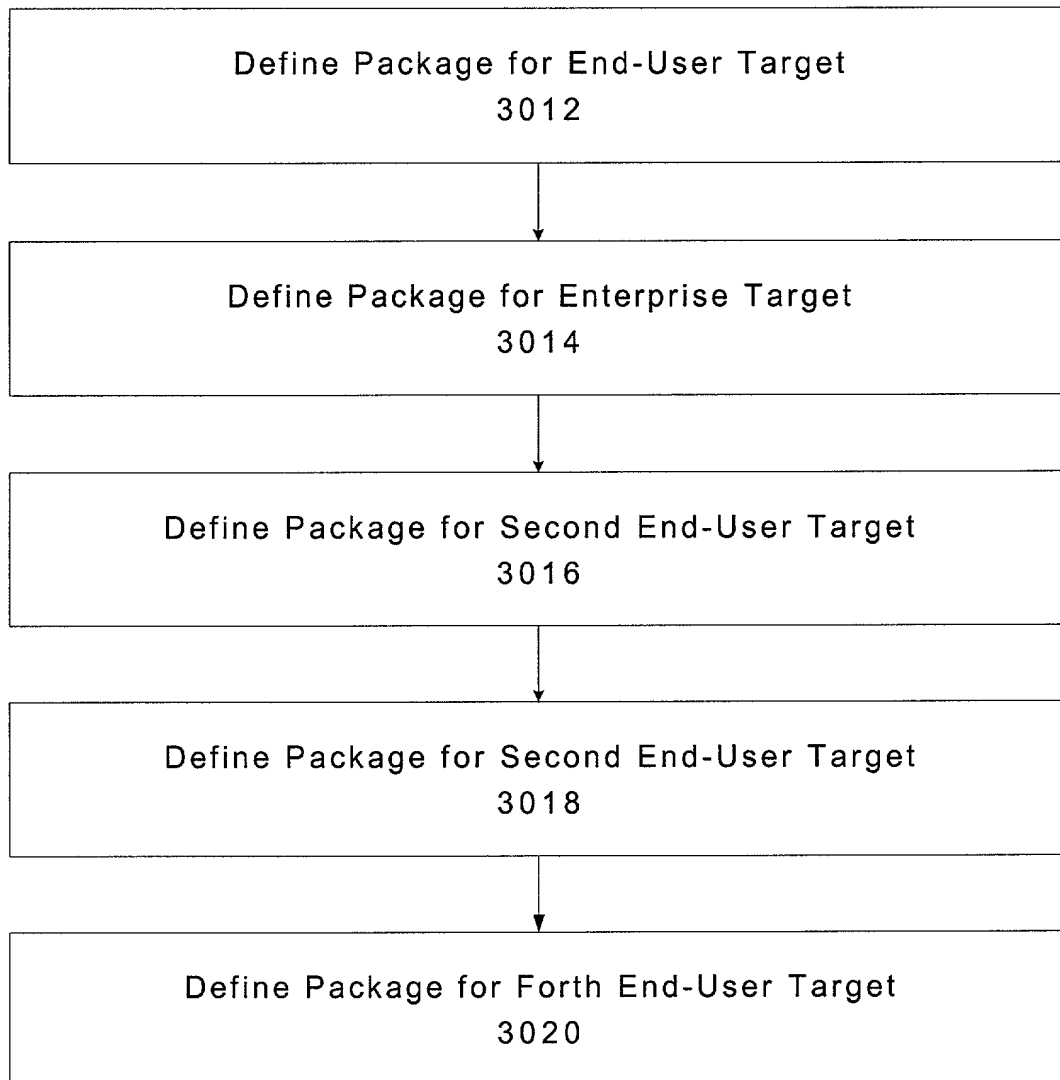
3000B



Server Definition

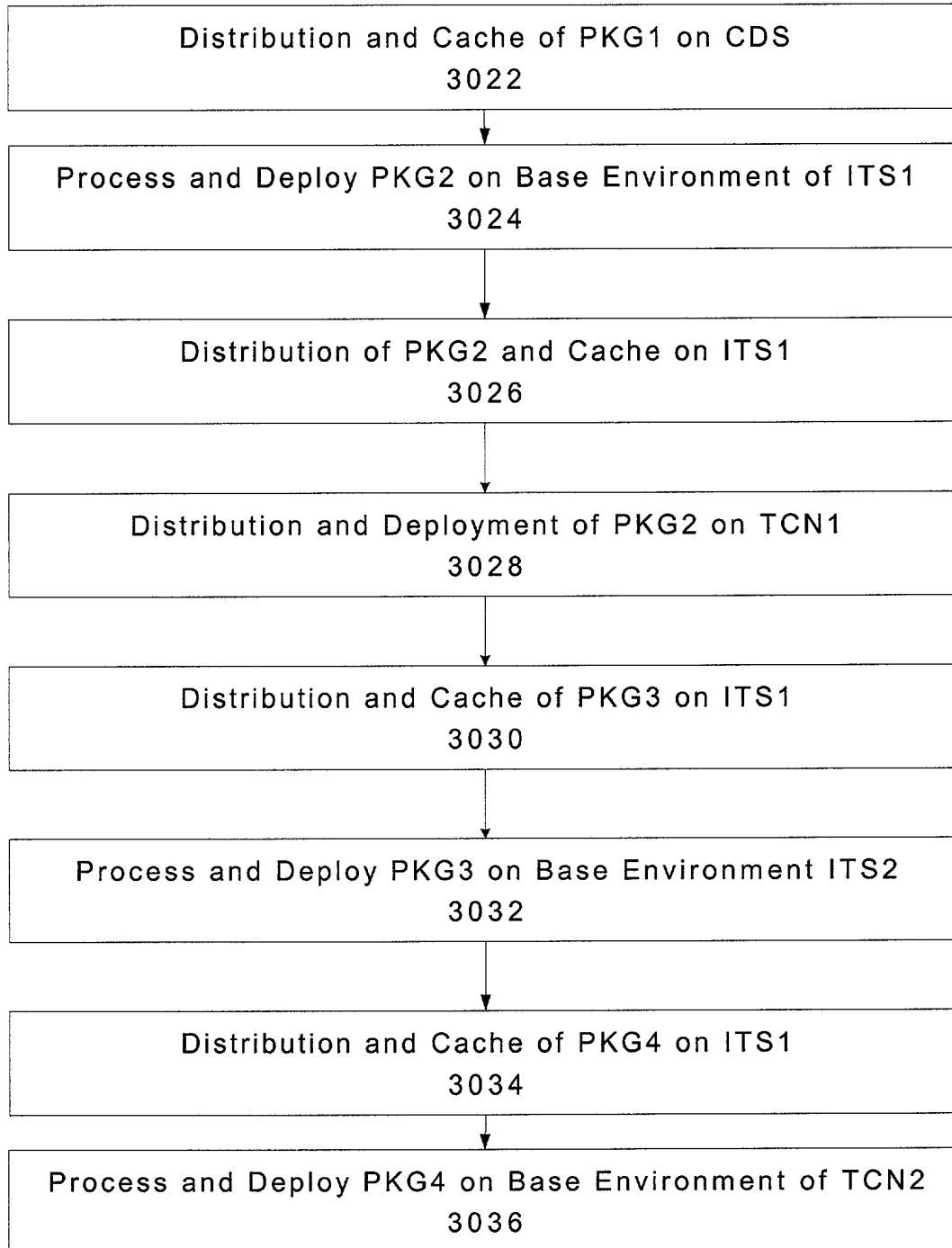
Figure 30B

3000C



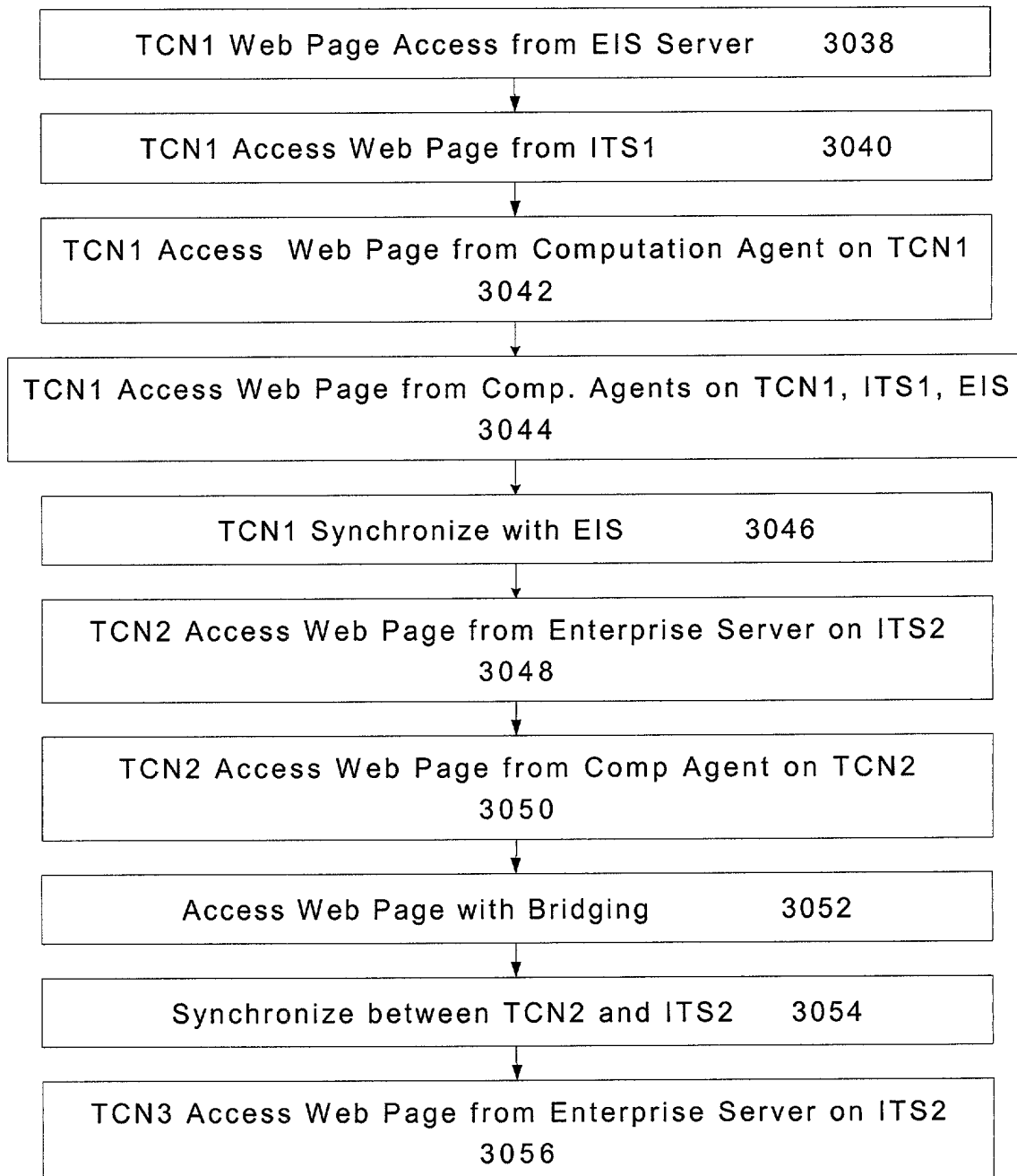
Define Packages/Applications  
Figure 30C

3000D



Distributing to Computational Environments

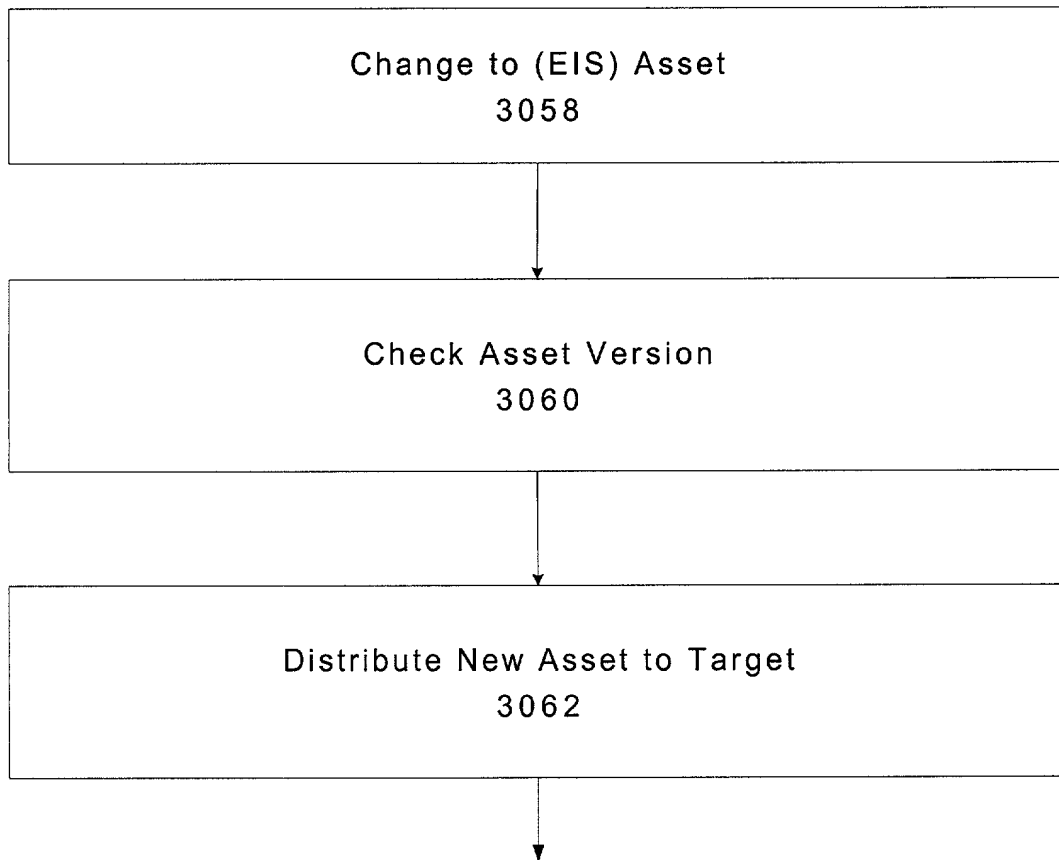
Figure 30D



Distributed Execution of Assets

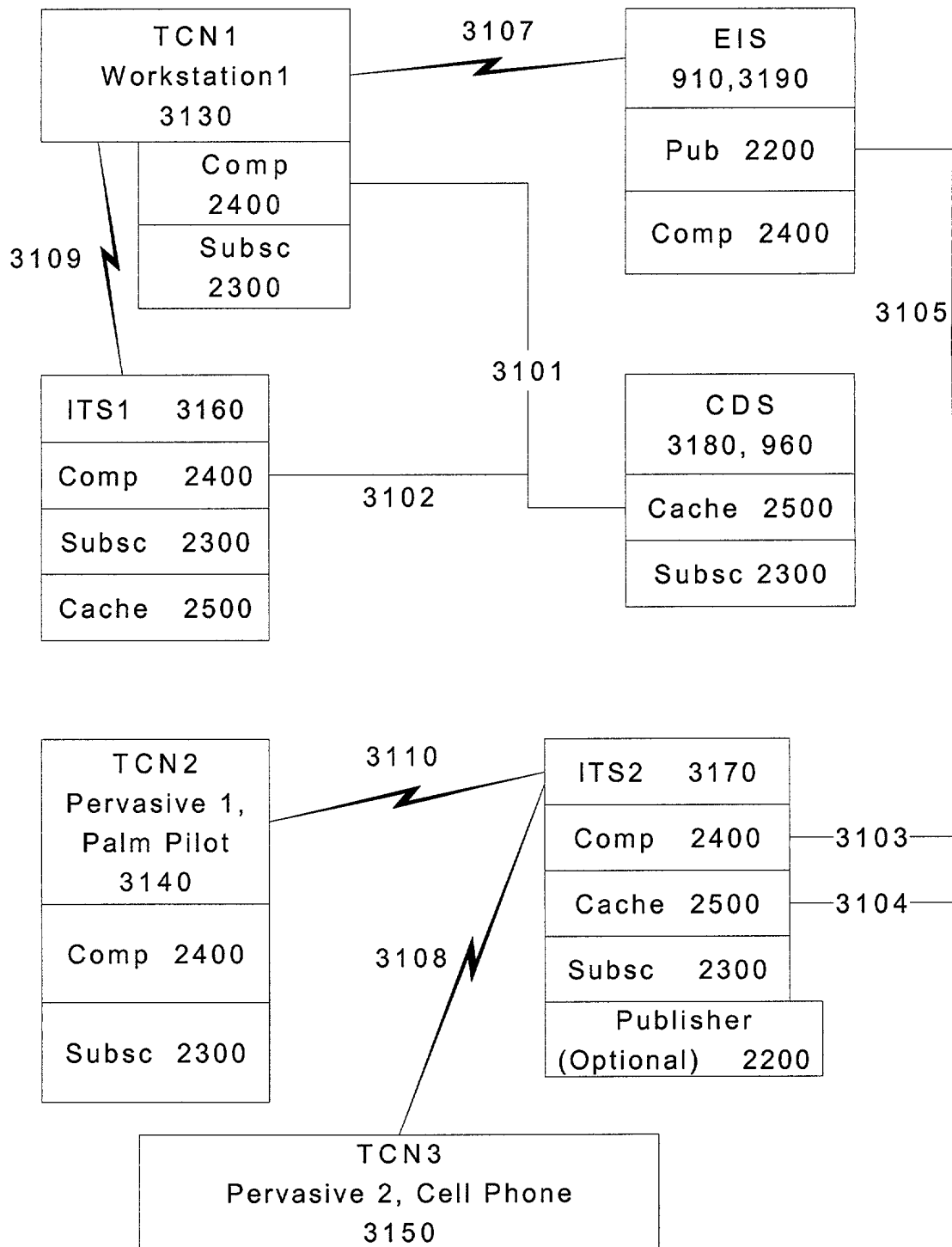
Figure 30E

3000F



Distribution of Current Assets  
Figure 30F

3100

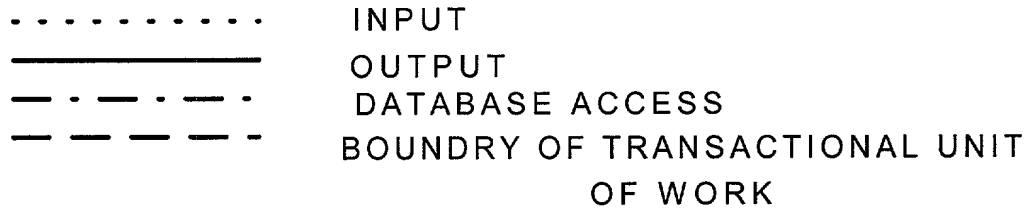


Example Network Connections and Asset Distribution

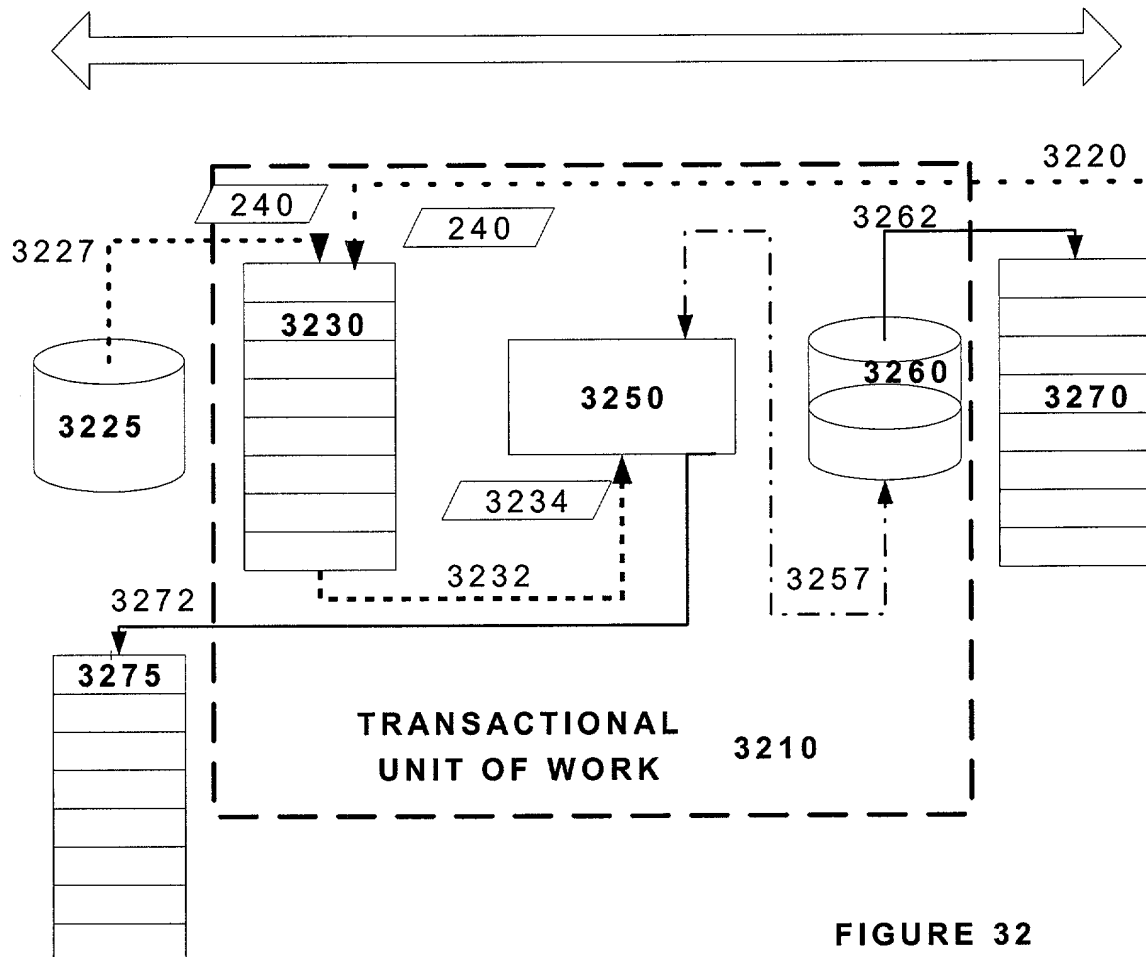
Figure 31



**3200**

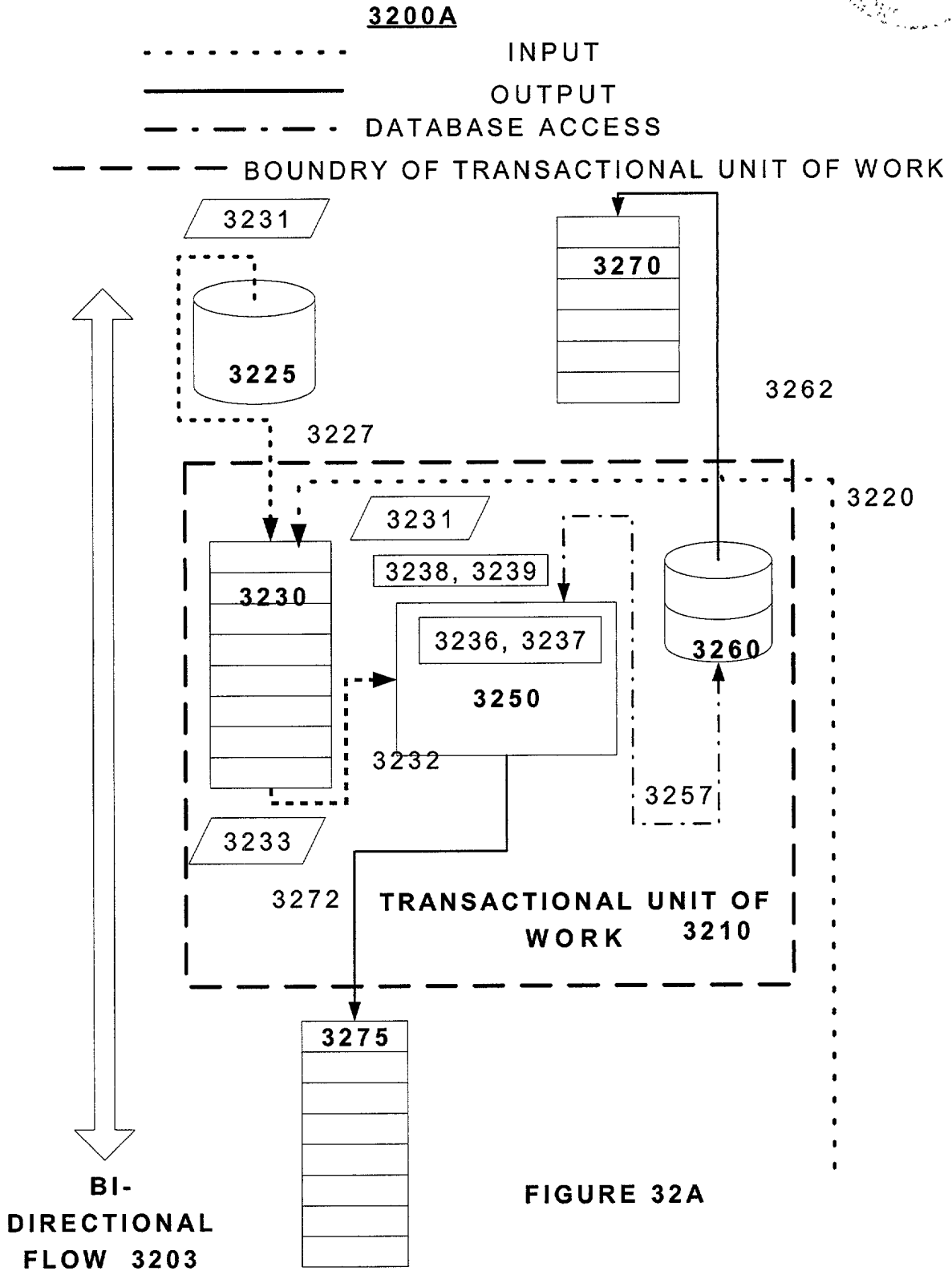


**BI-DIRECTIONAL FLOW 3203**



**FIGURE 32**

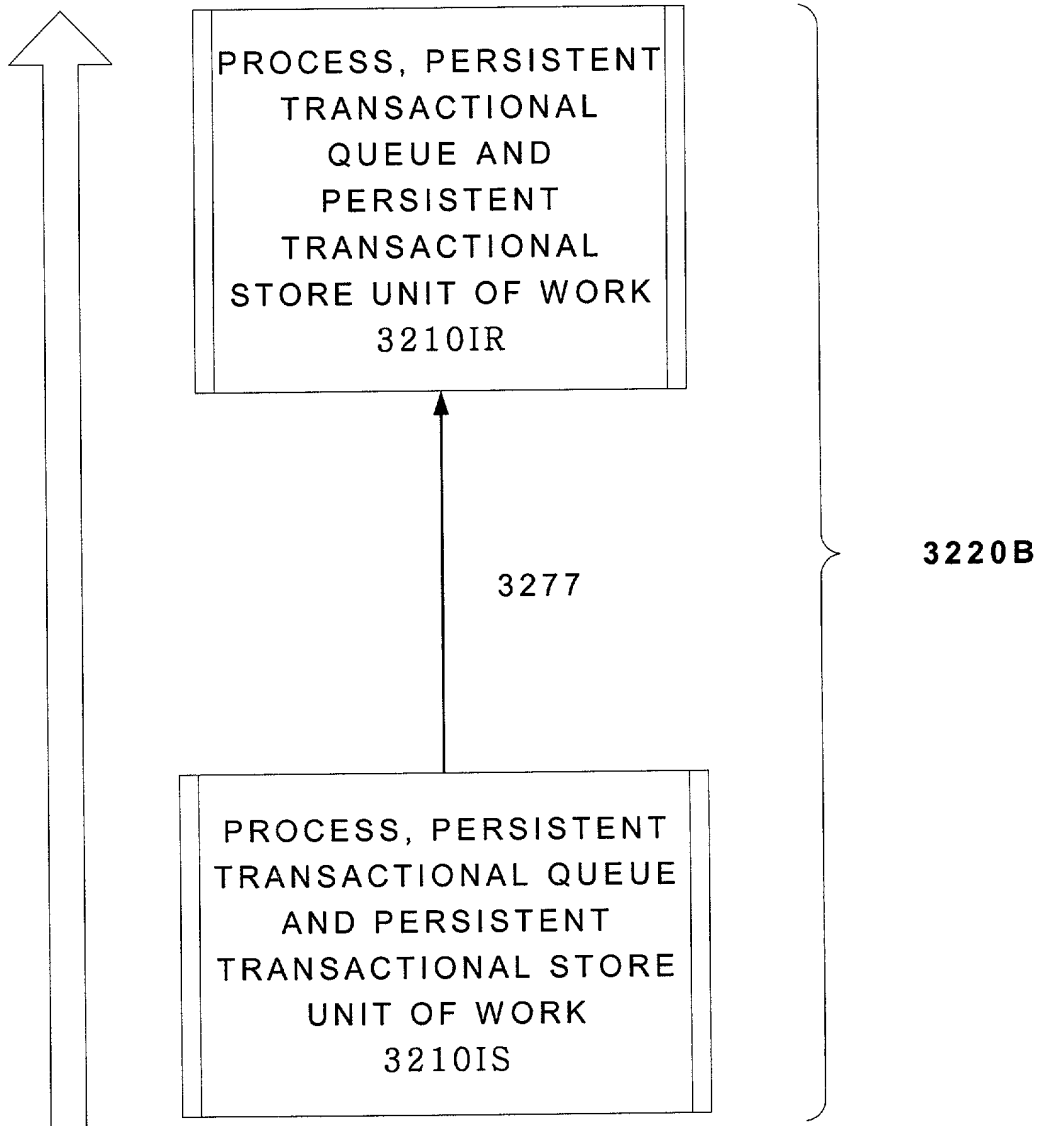






**3200B**

Control  
Flow  
3205



**FIGURE 32B**

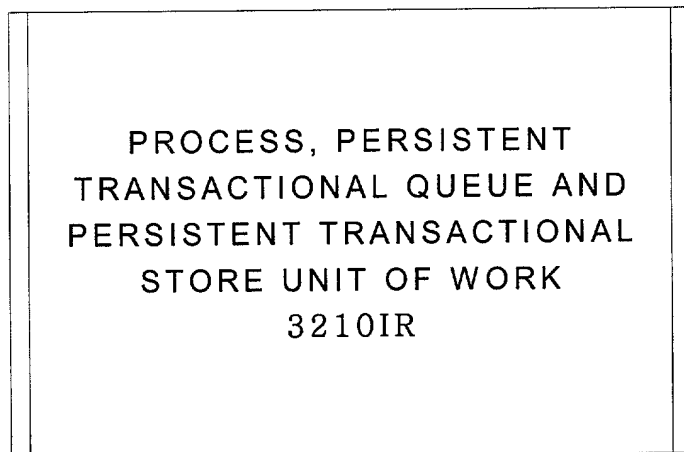


**3200C**

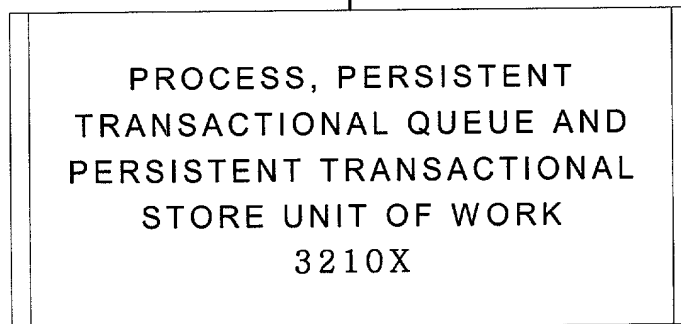
Control

Flow

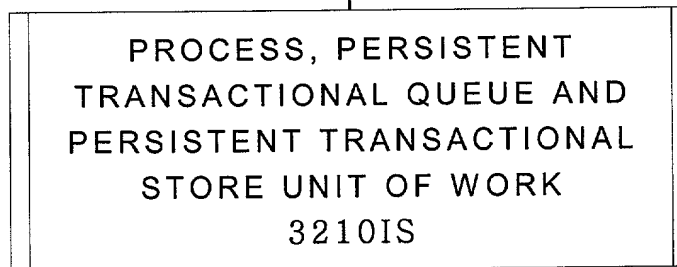
3205



3279



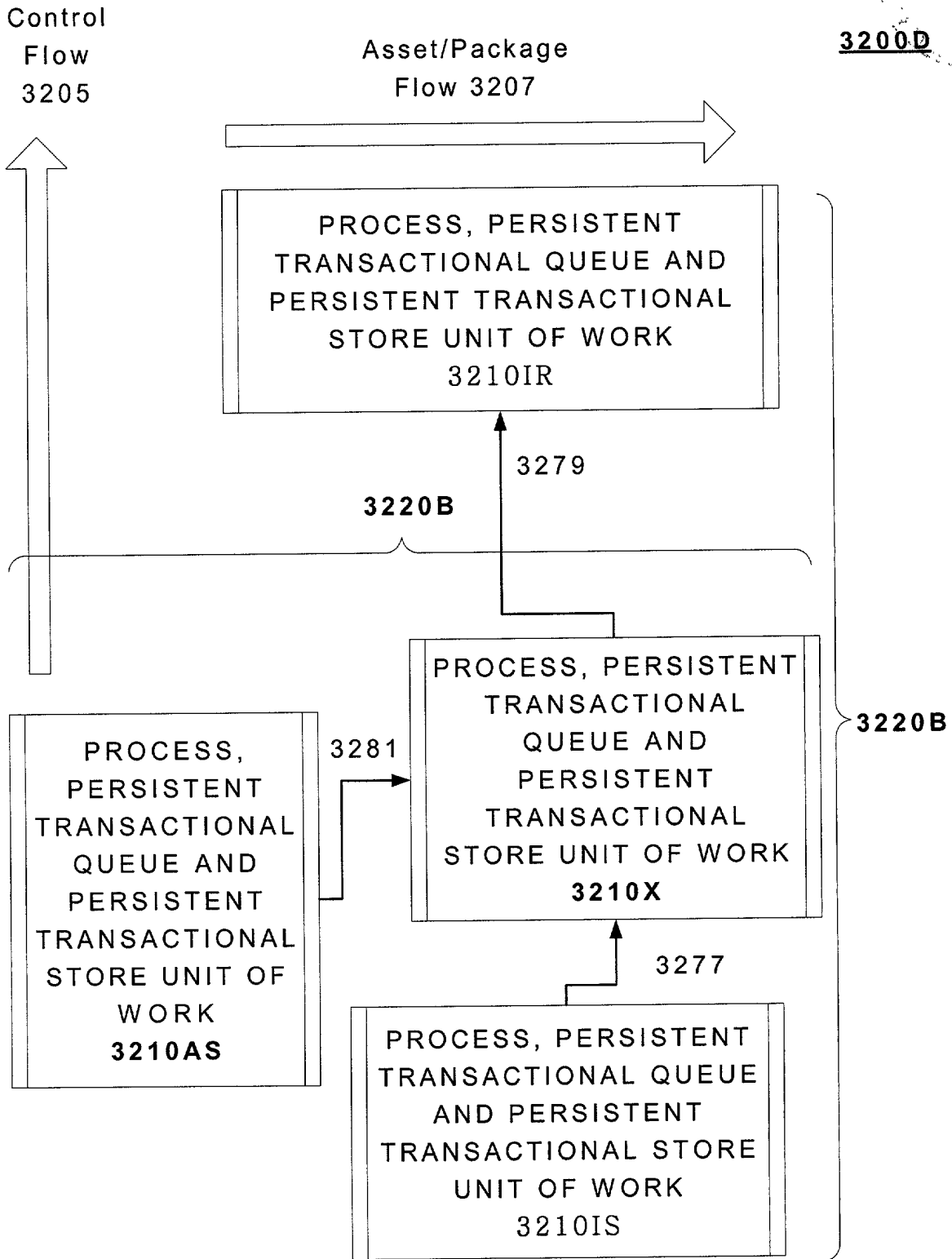
3277



**3220B**

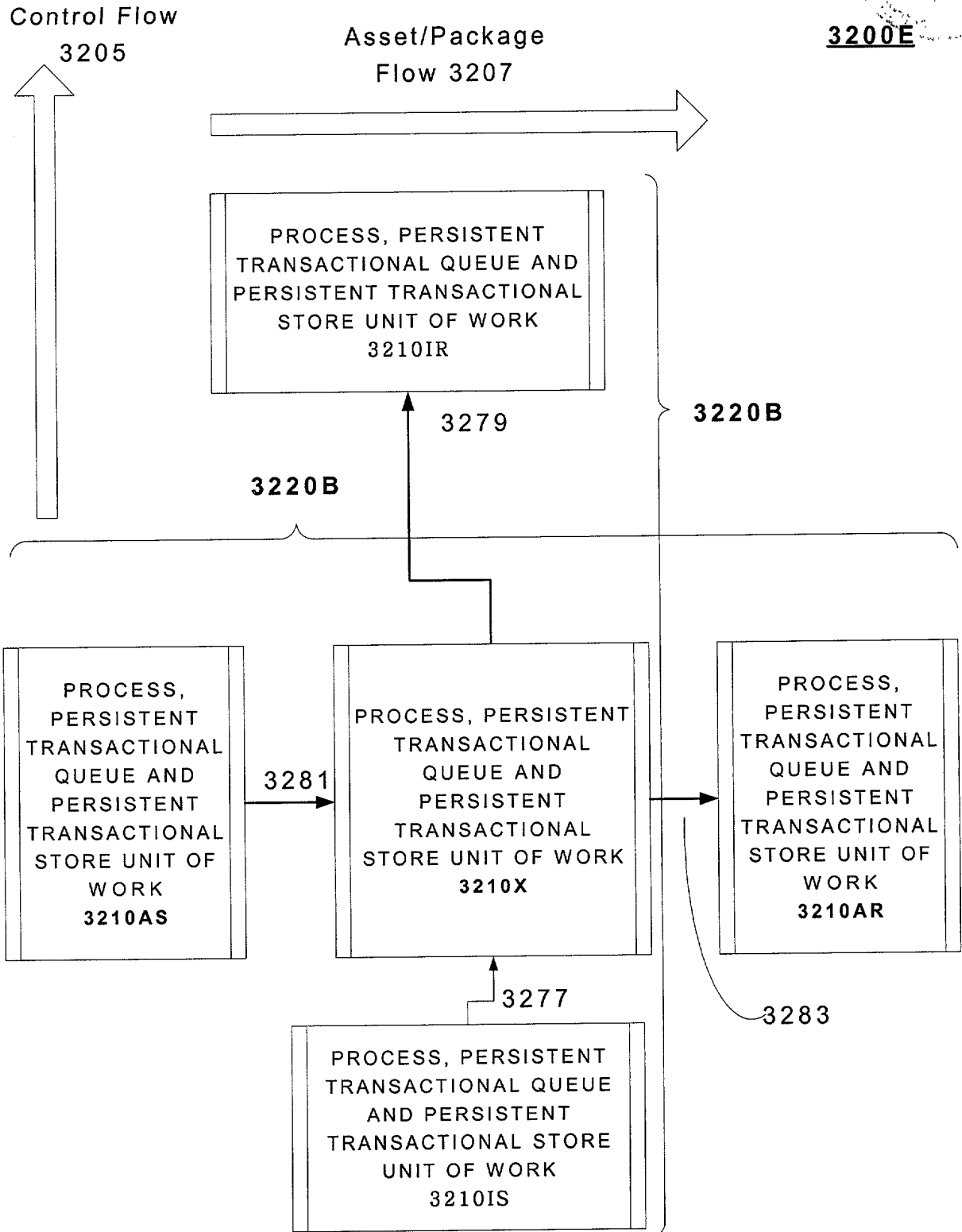
**FIGURE 32C**

**3200D**



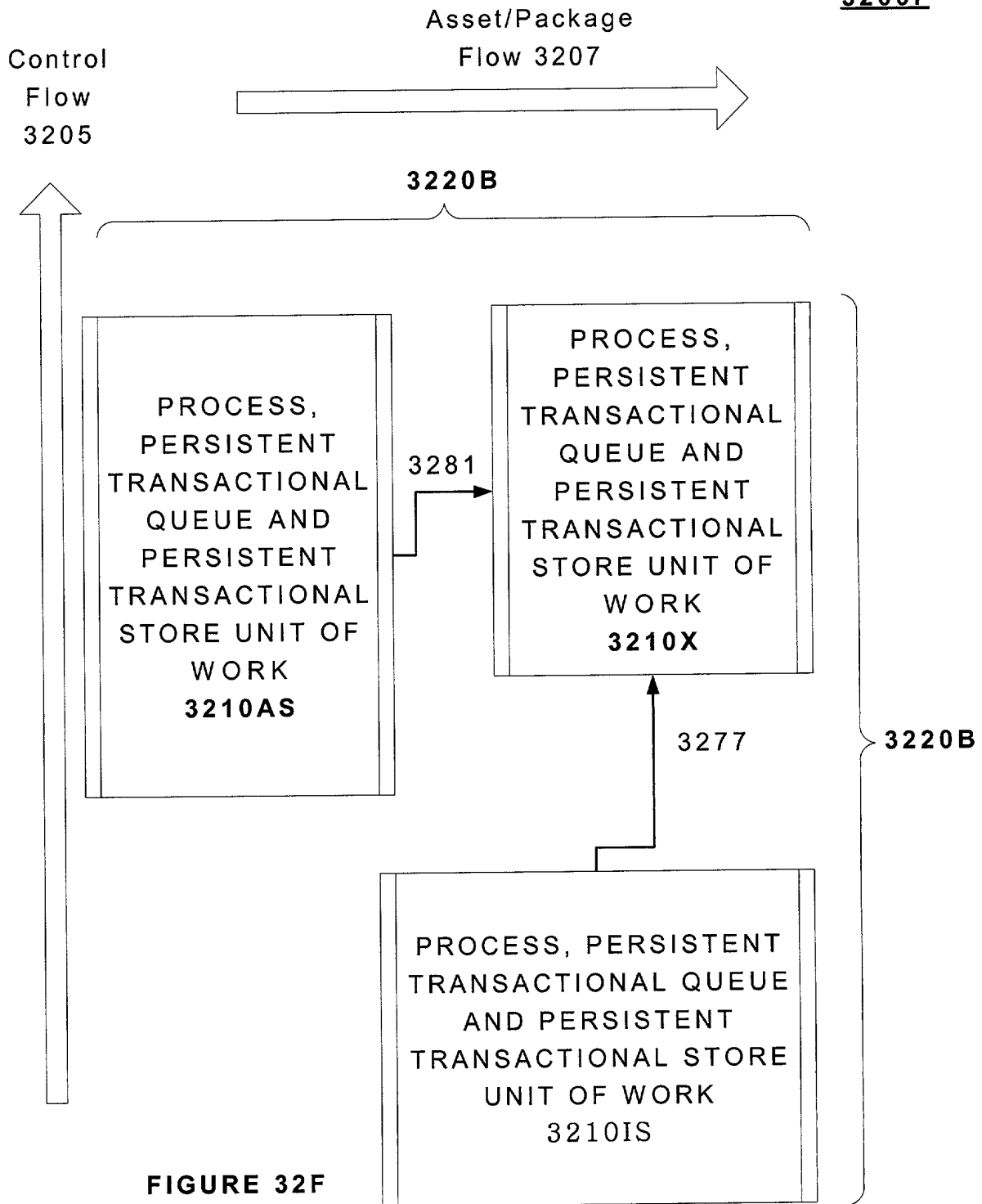
**FIGURE 32D**

**3200E**



**FIGURE 32E**

**3200F**



**FIGURE 32F**

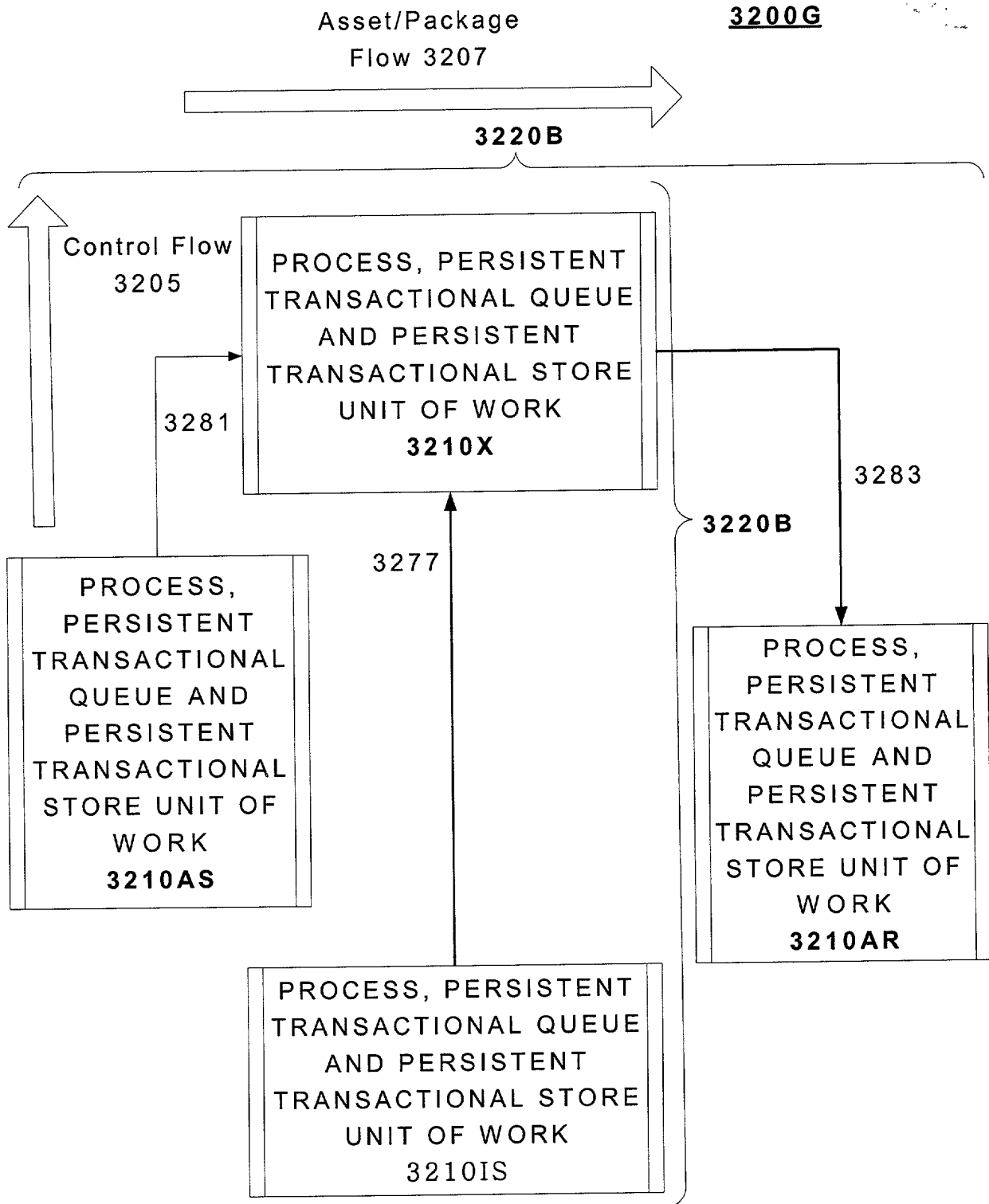
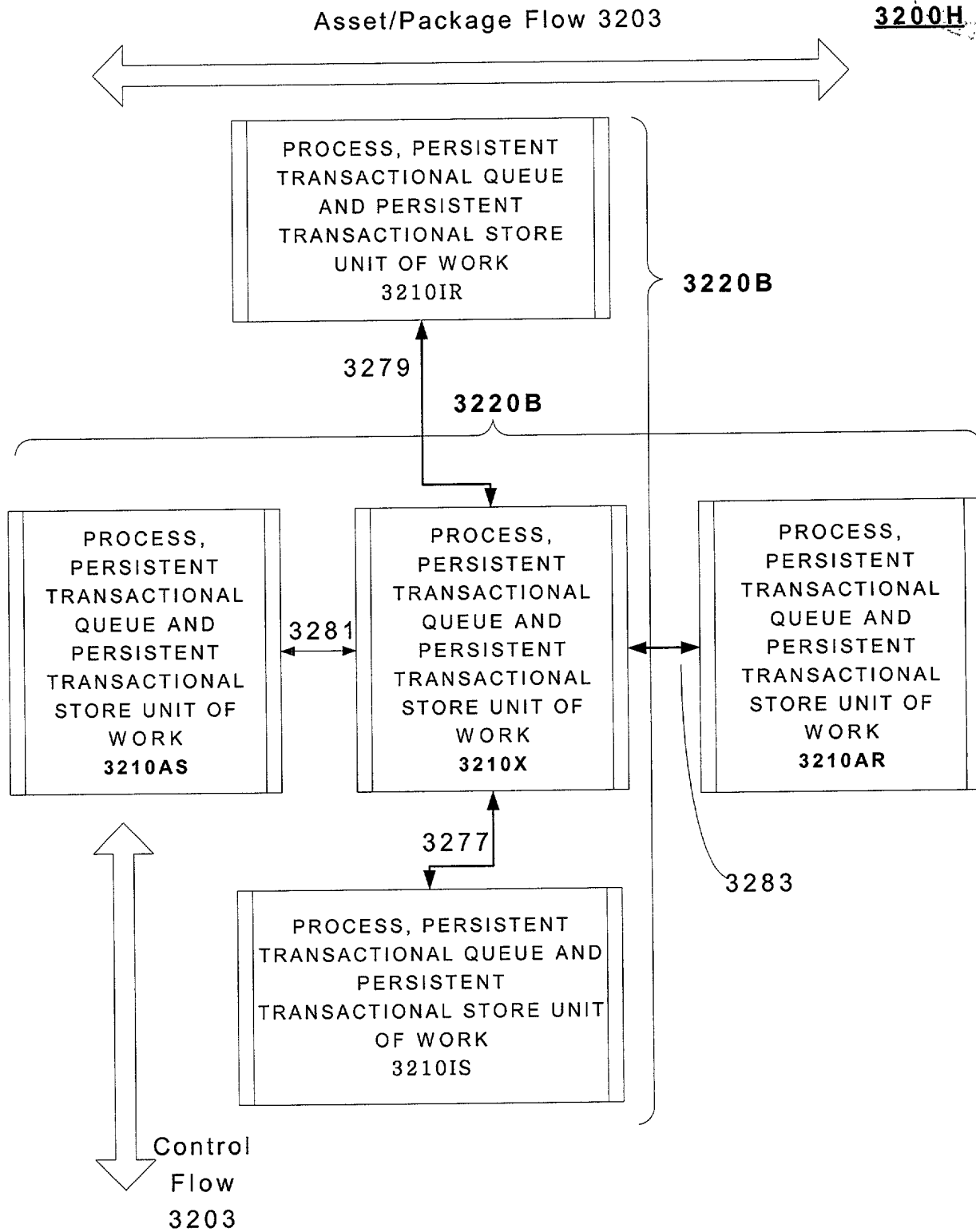


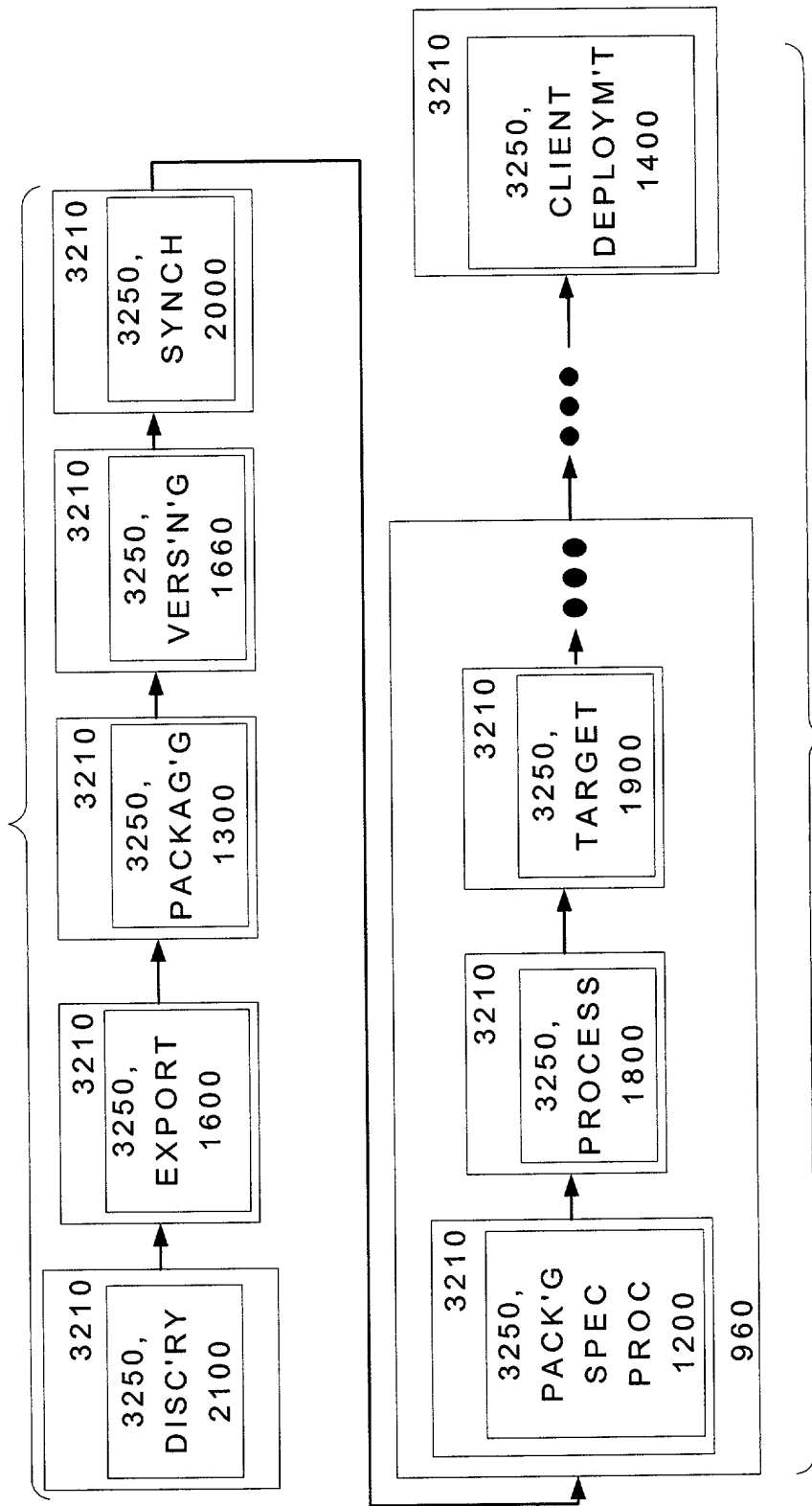
FIGURE 32G





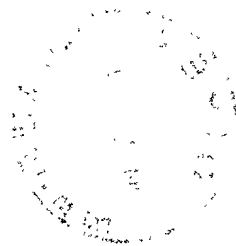
3300

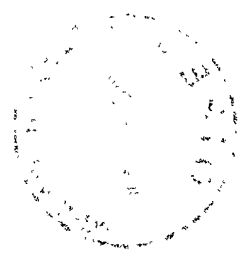
3220B



3220B

FIGURE 33





3300A

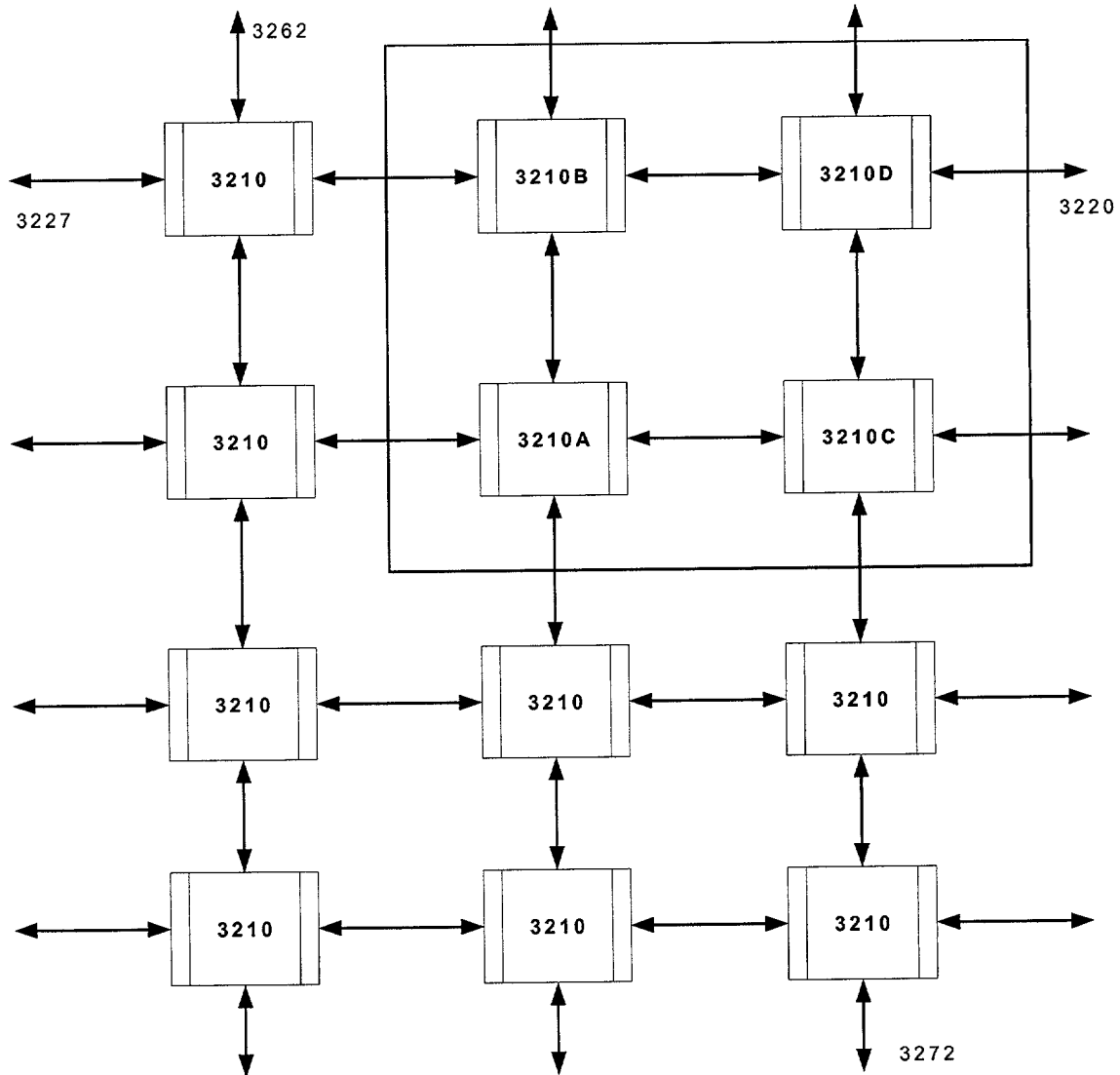


FIGURE 33A

**Figure 1 - Collaboration System Configuration**

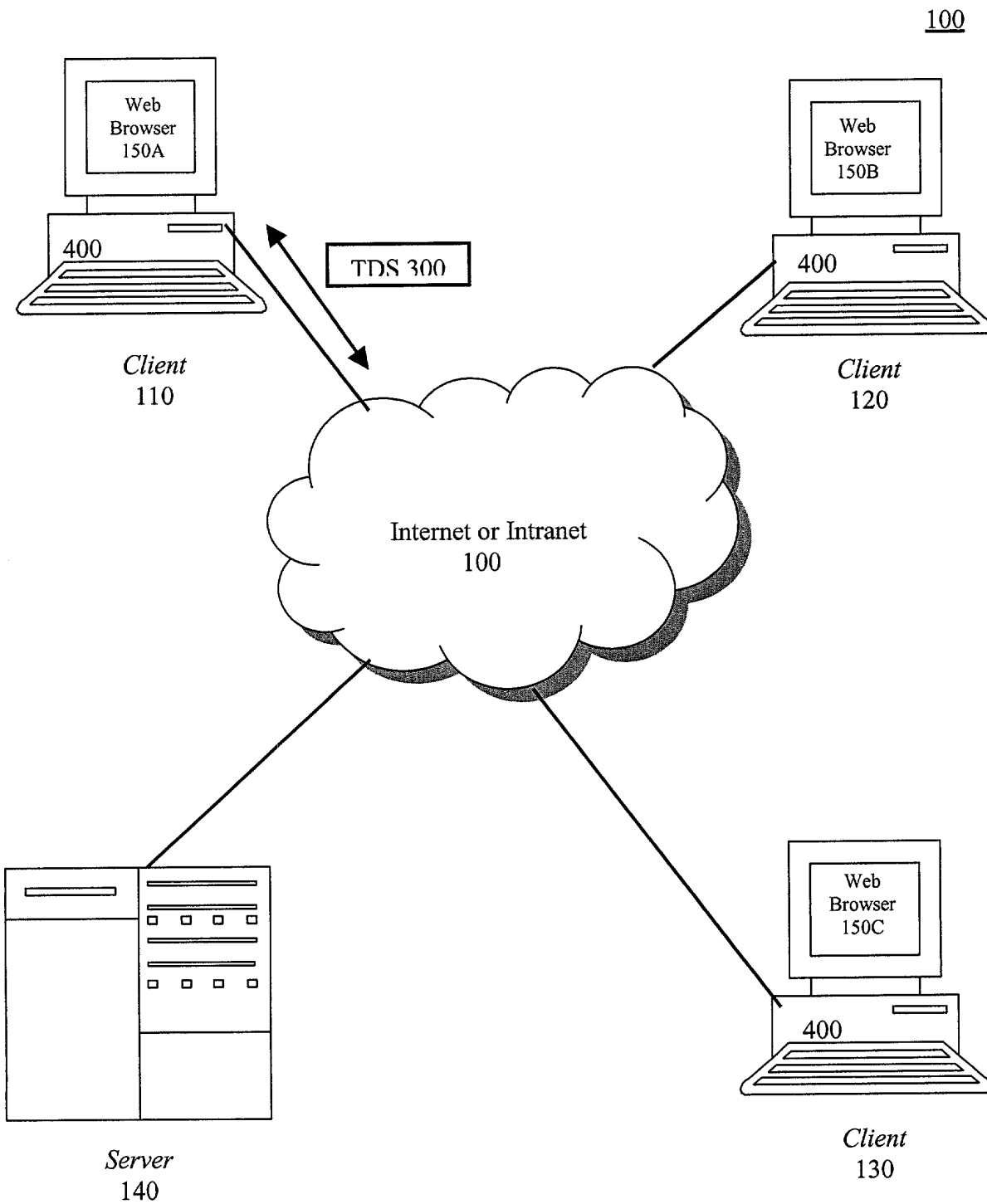
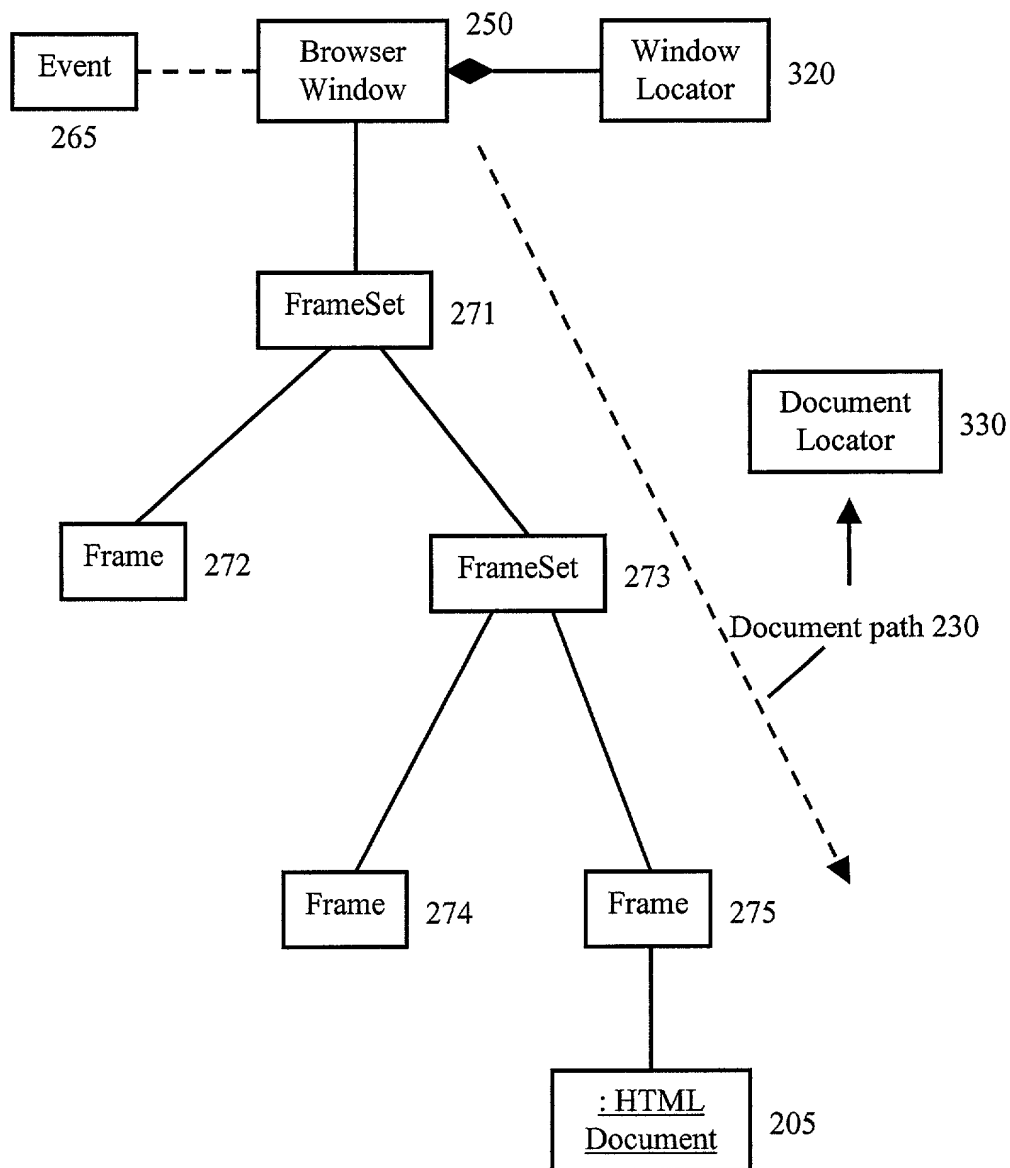


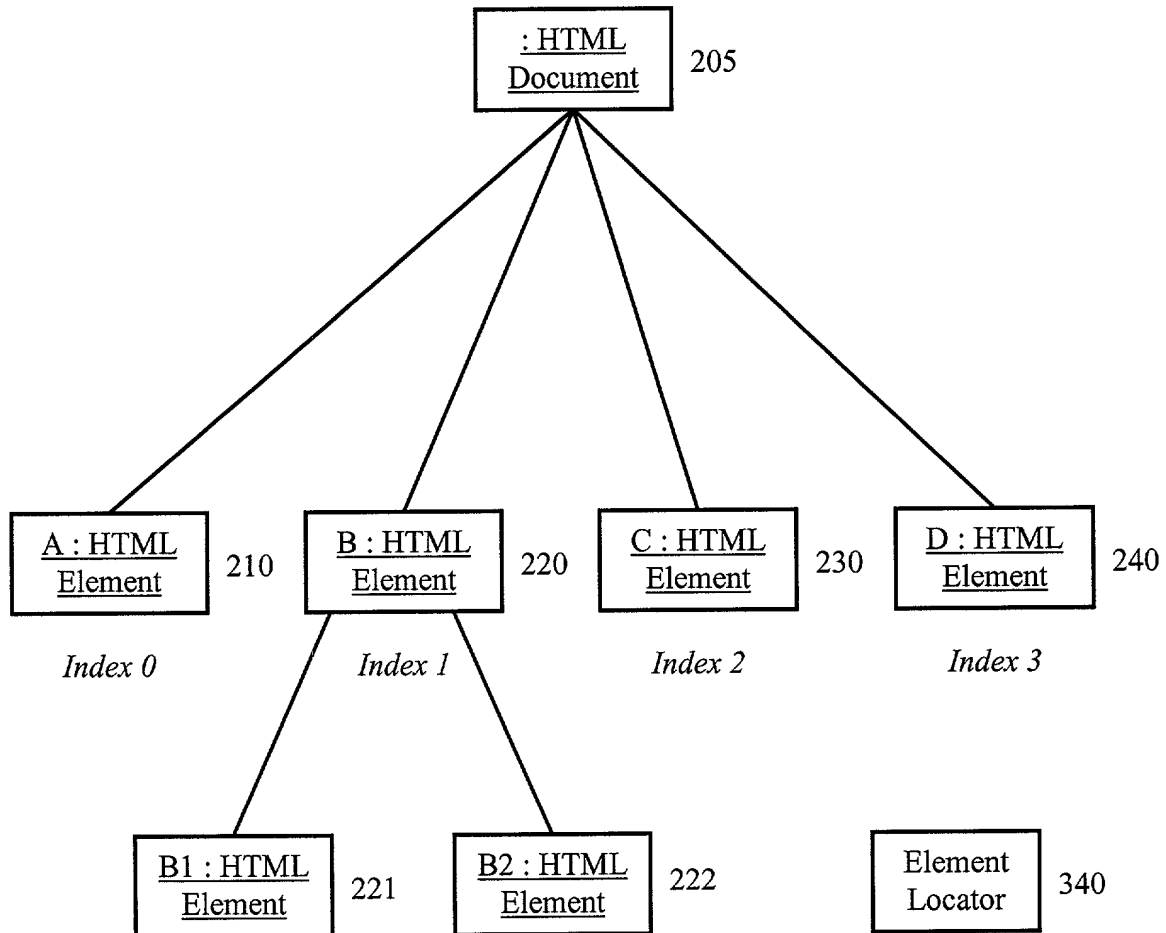
Figure 2A – Window Locator and Document Path

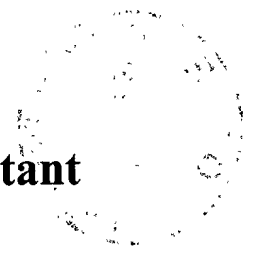
290



**Figure 2B – DOM/DHTML Structure in a Web Browser**

200





## Figure 2C – Element Locator, showing optional mutant web page support

340

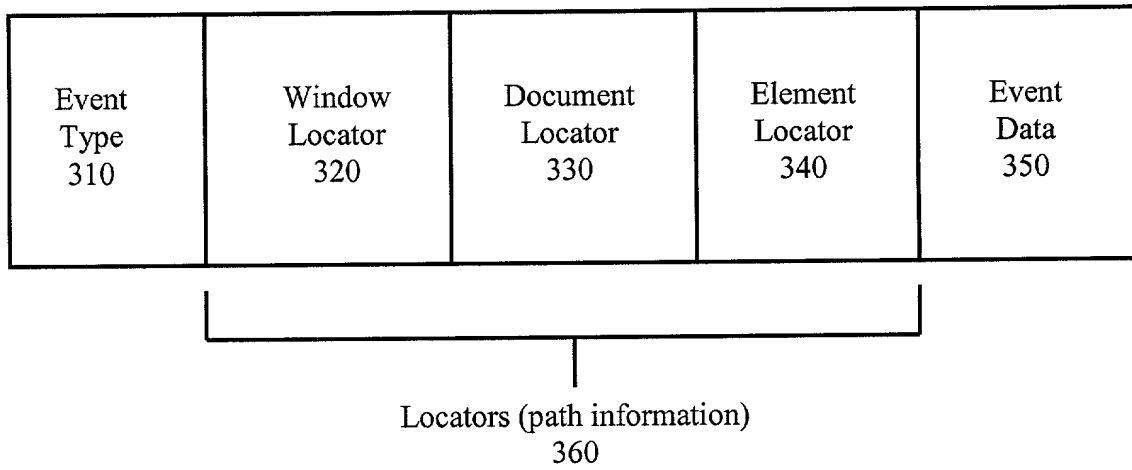
Element tagName and index 341	Mutant web page hash value 342
--	--------------------------------------

Chen et al. 09/944,062



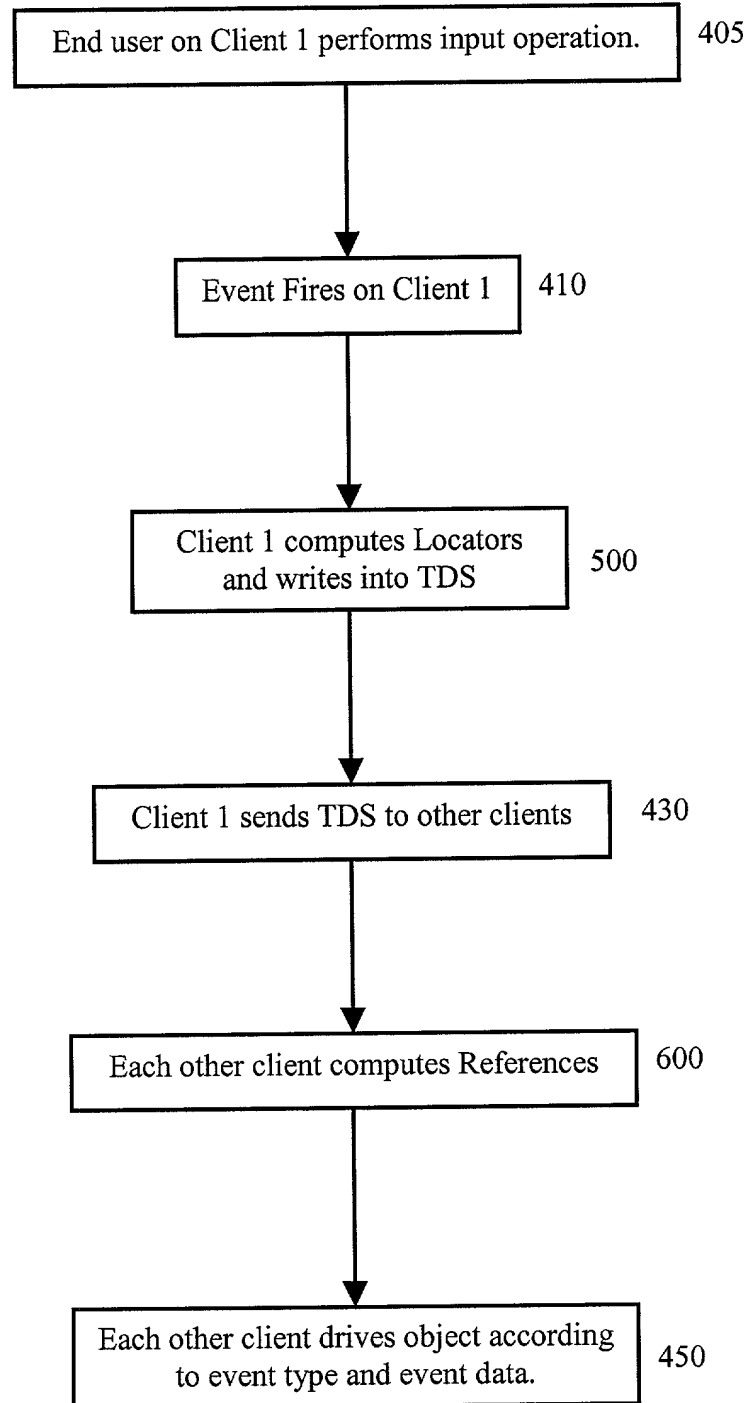
**Figure 3 – Transfer Data Structure (TDS)**

**Transfer Data Structure (TDS)**  
300



**Figure 4 -- Collaboration System Flow**

400





**Figure 5 – Computing Locators and Writing into TDS**

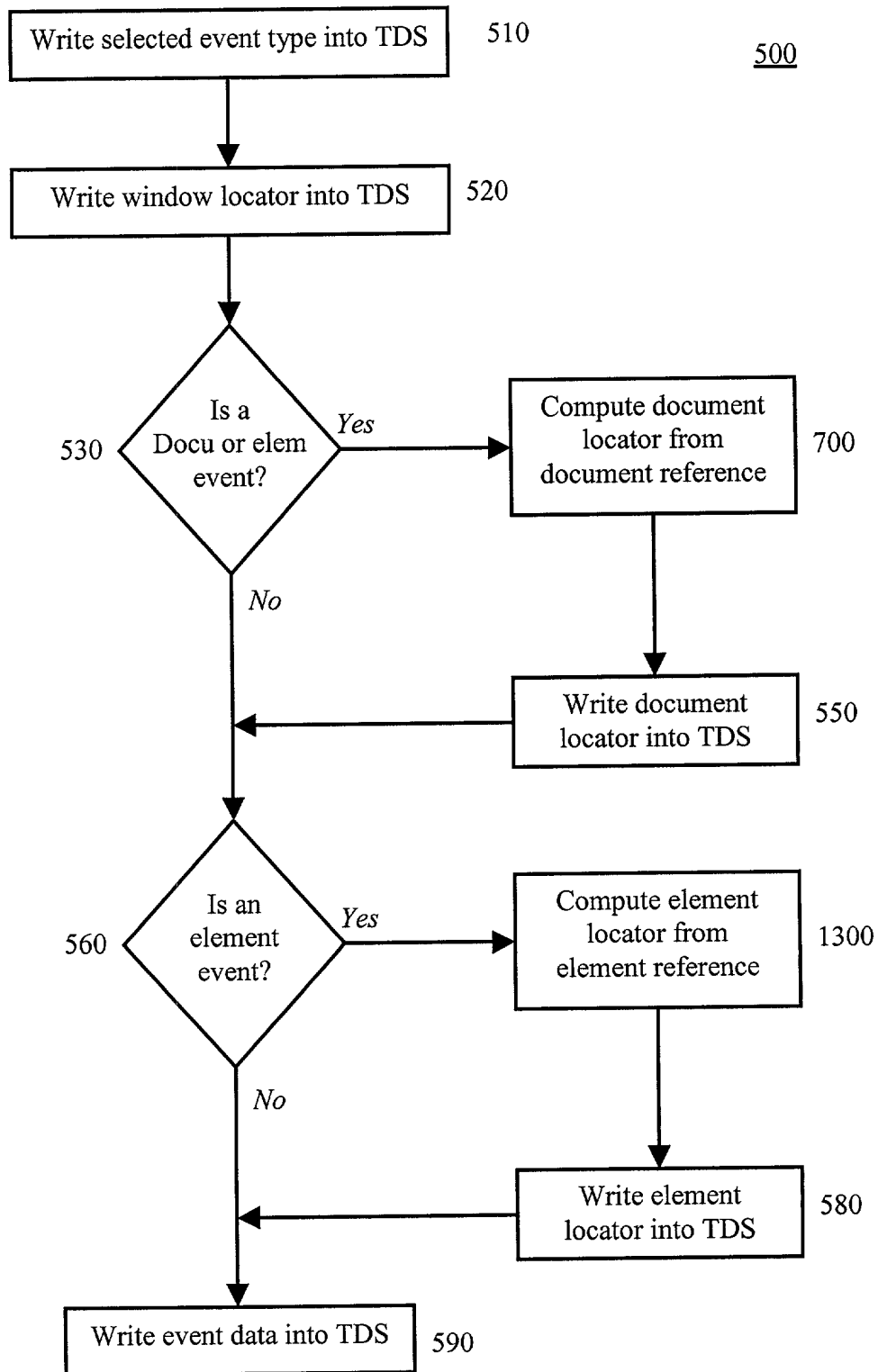
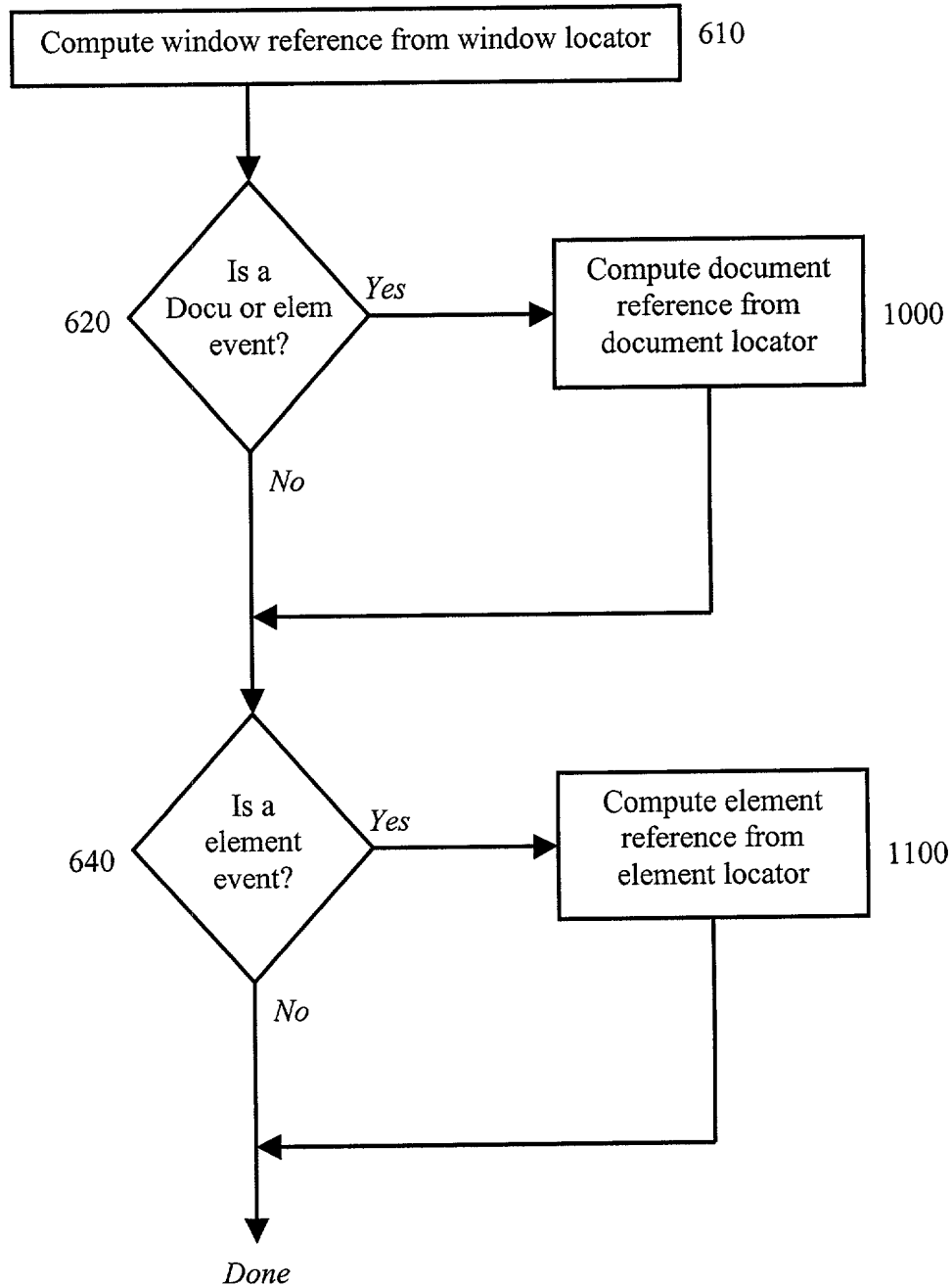


Figure 6 – Computing References



**Figure 7 – Compute Document Locator from Document Reference**

700

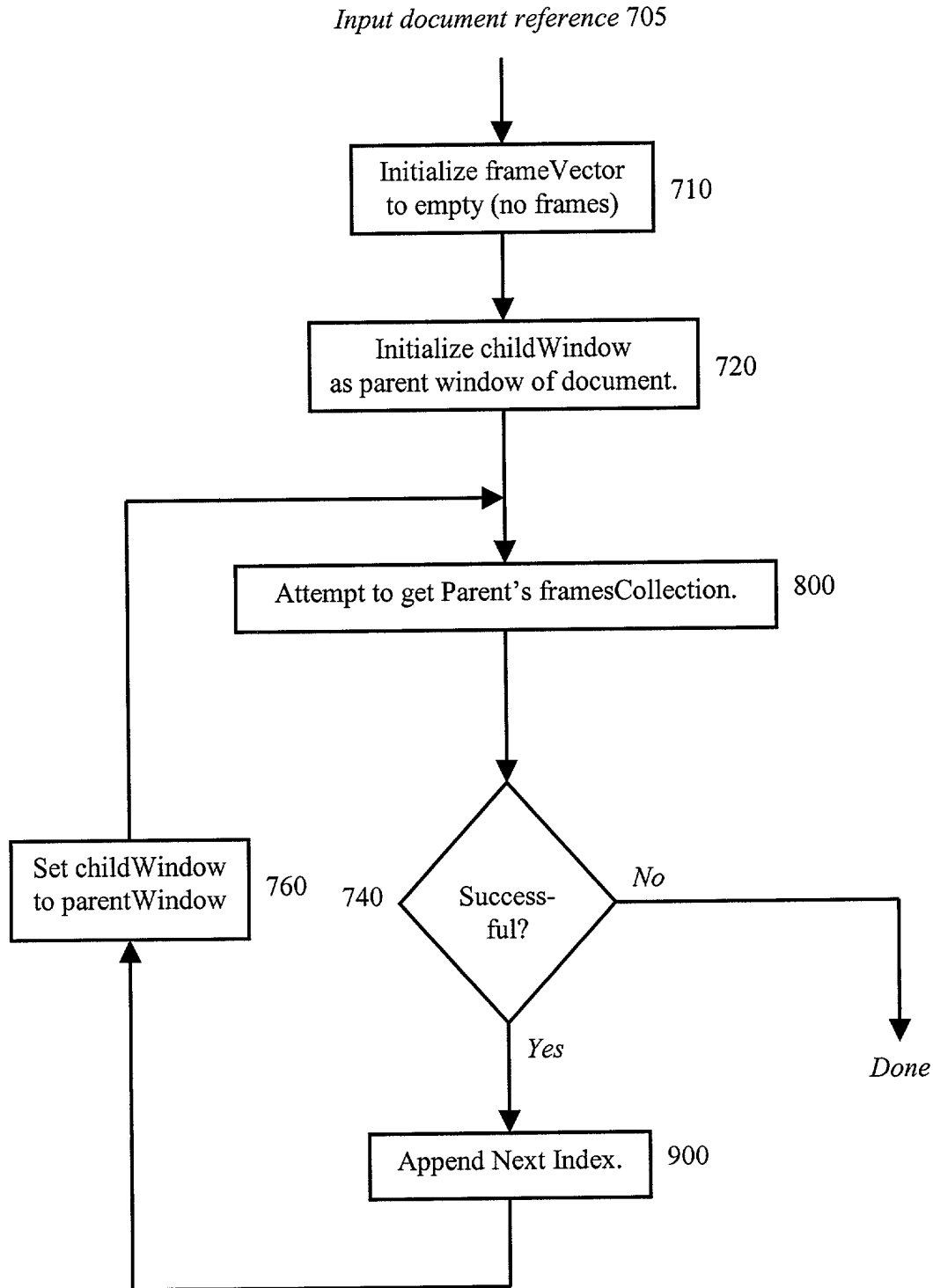


Figure 8 – Attempt to Get Parent's framesCollection

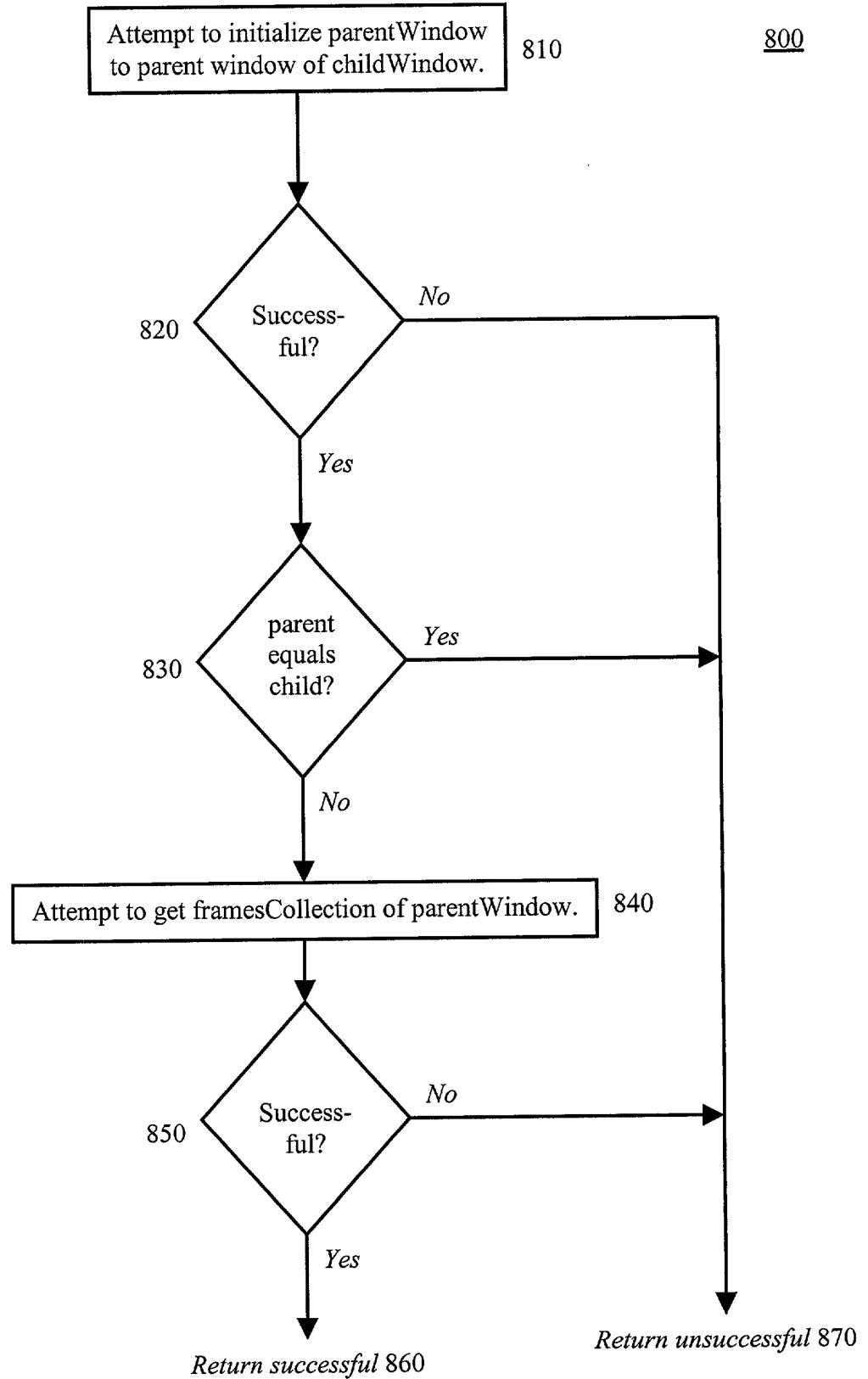
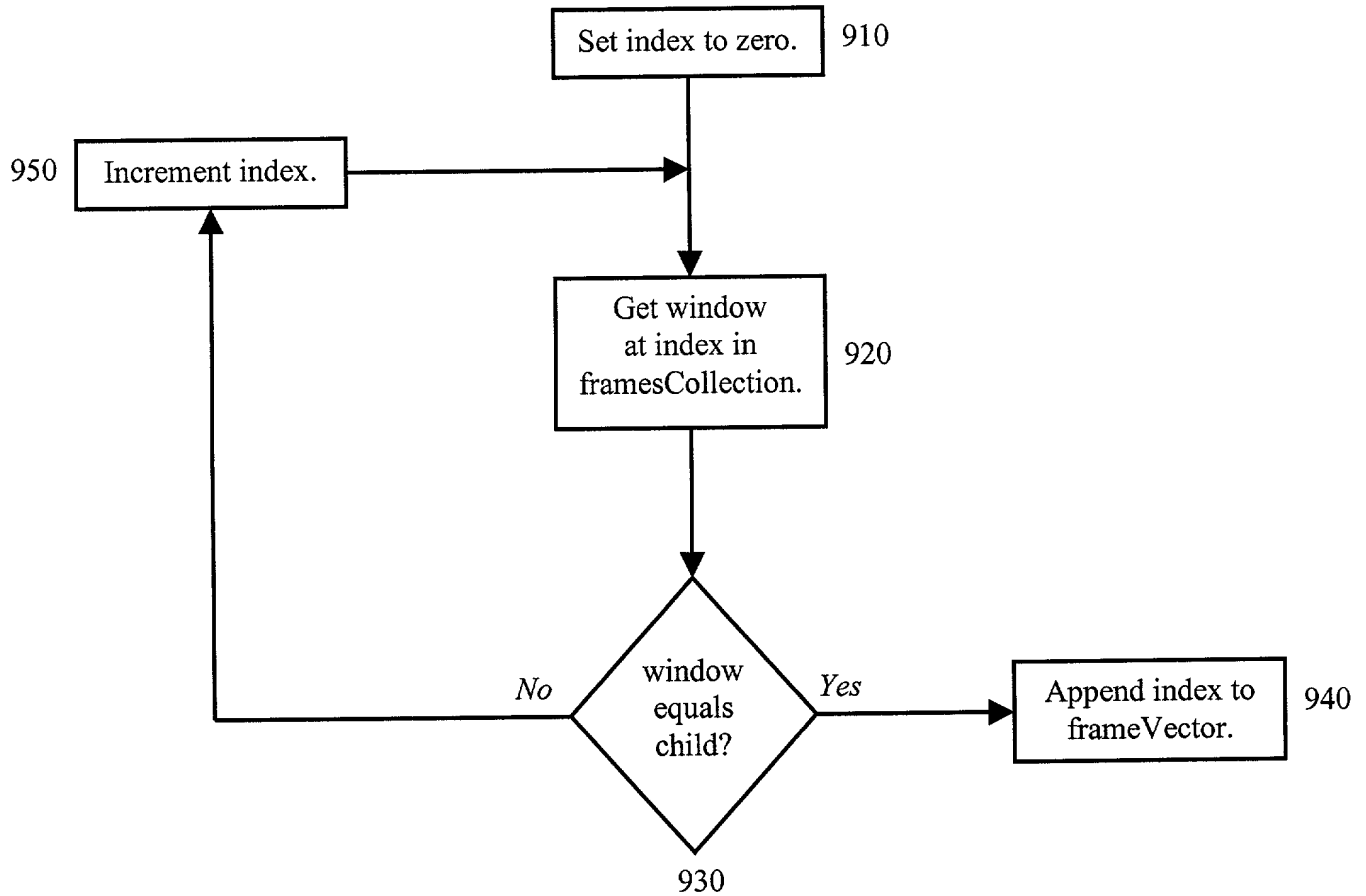


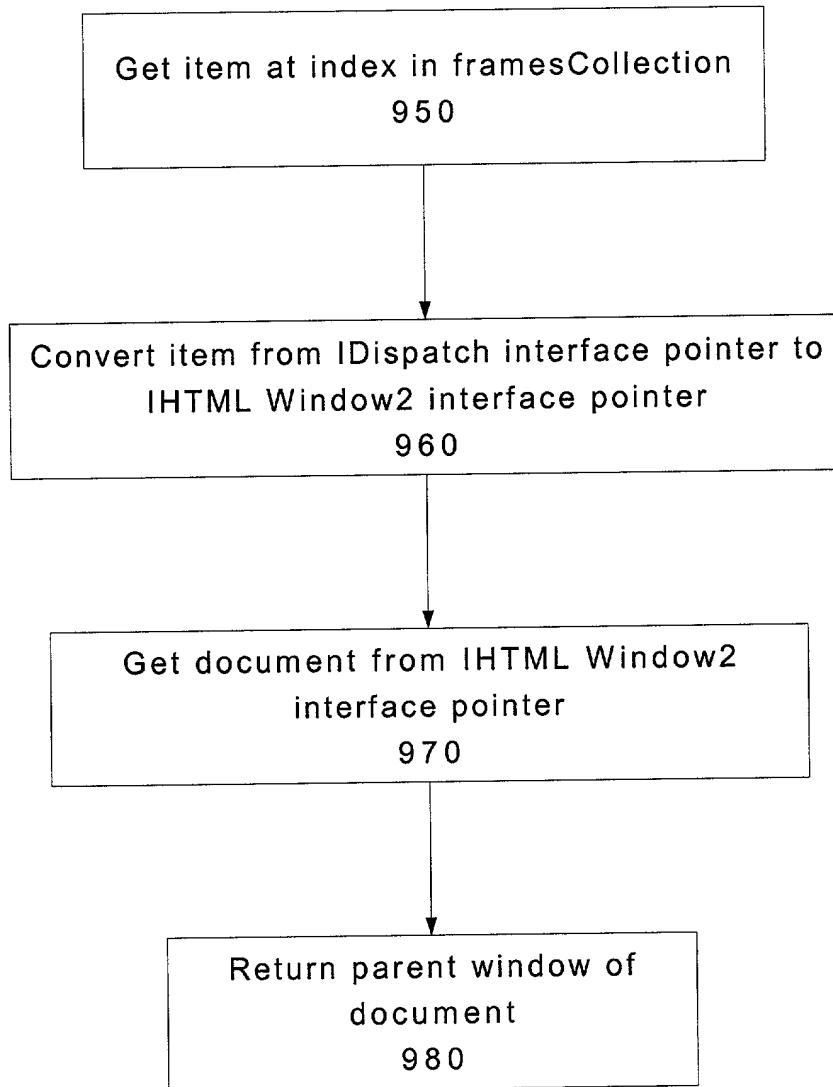
Figure 9A – Append Next Index

900





One Embodiment Of  
920



Compute Window at Index in IE5  
FramesCollection

Figure 9B

## Figure 10 (Sheet 1 of 2) – Compute Document Reference from Document Locator

1000

*Input frameVector from Document Locator*

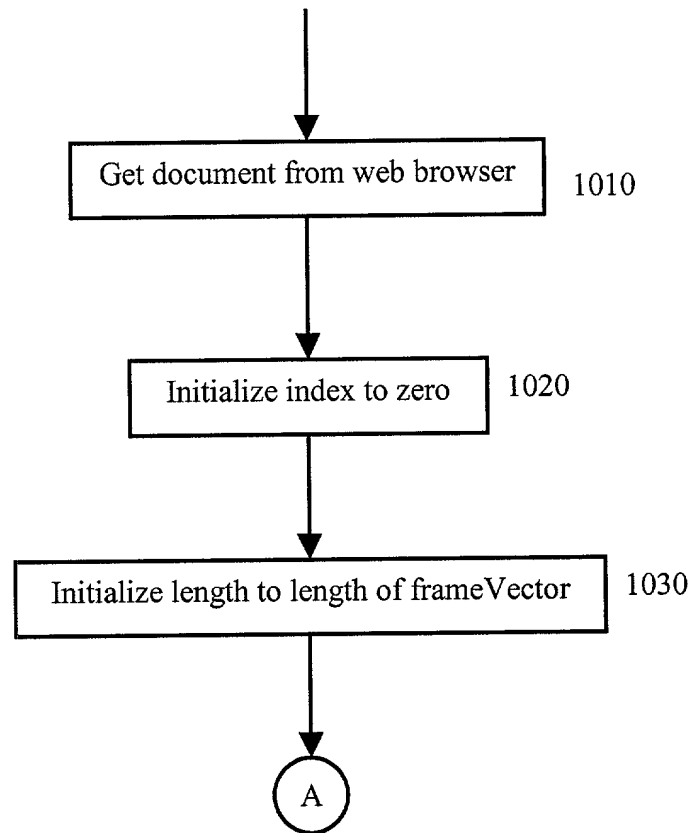
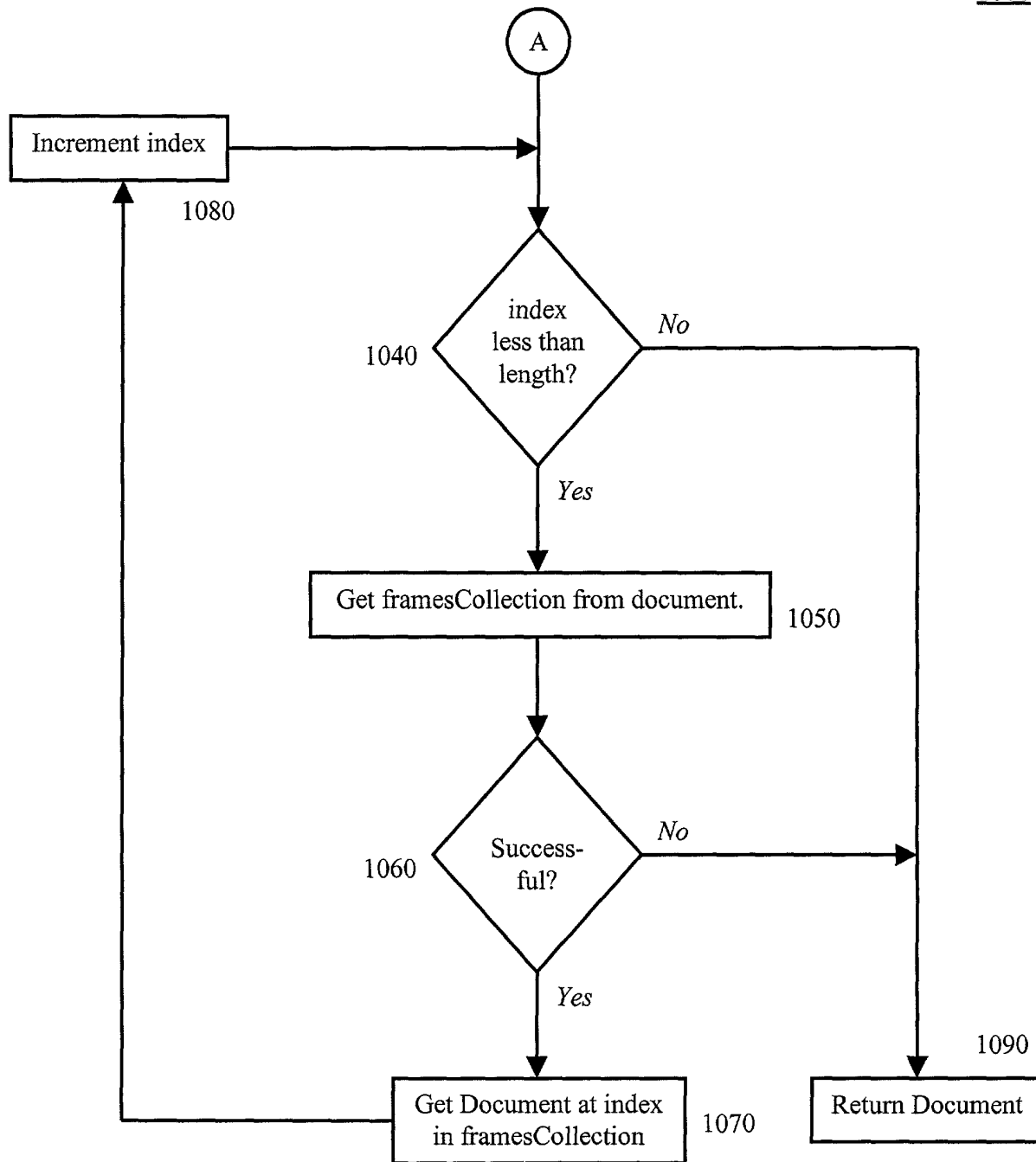


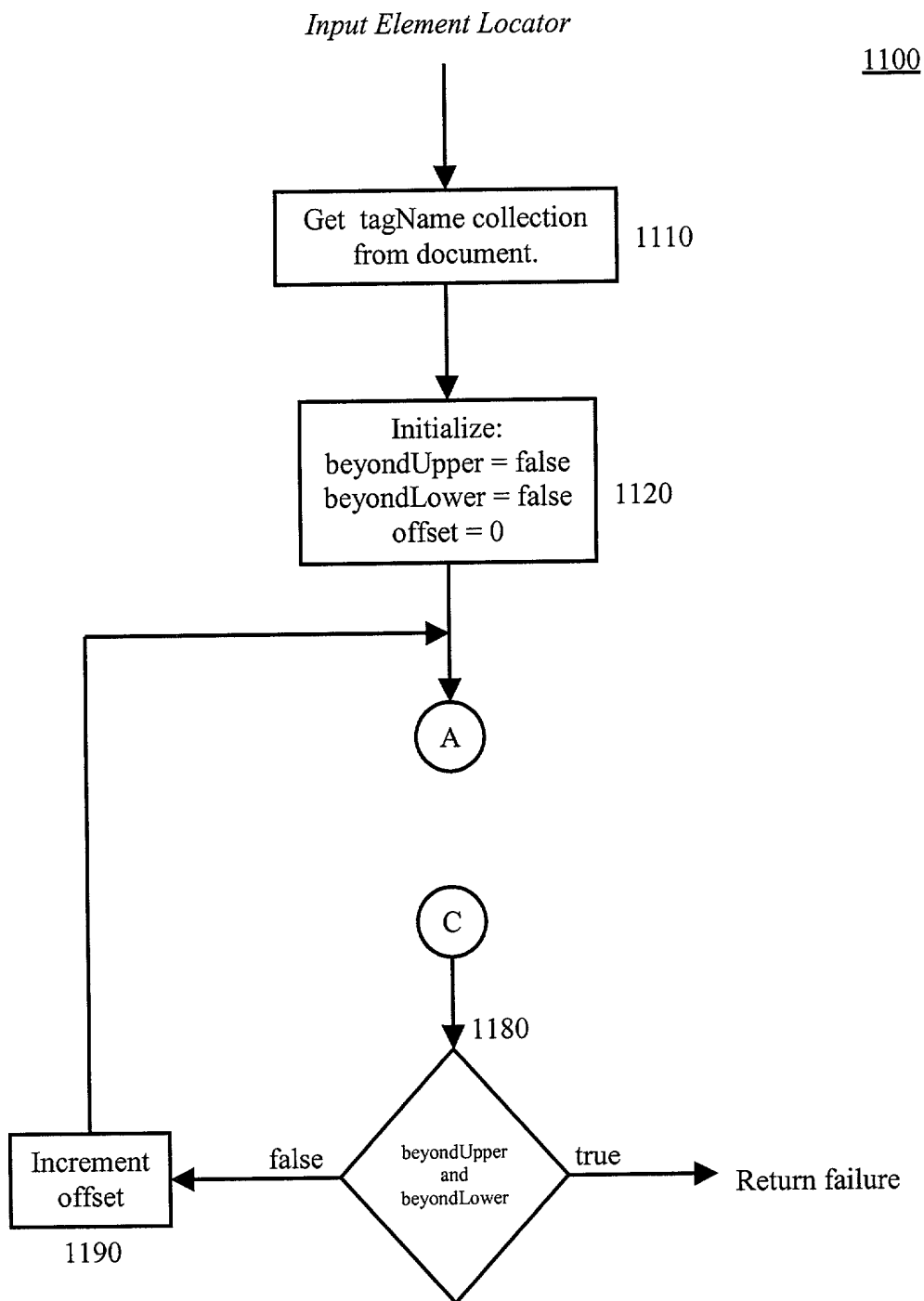
Figure 10 (Sheet 2 of 2)

1000





**Figure 11 (Sheet 1 of 3) – Compute Element Reference  
from Element Locator**



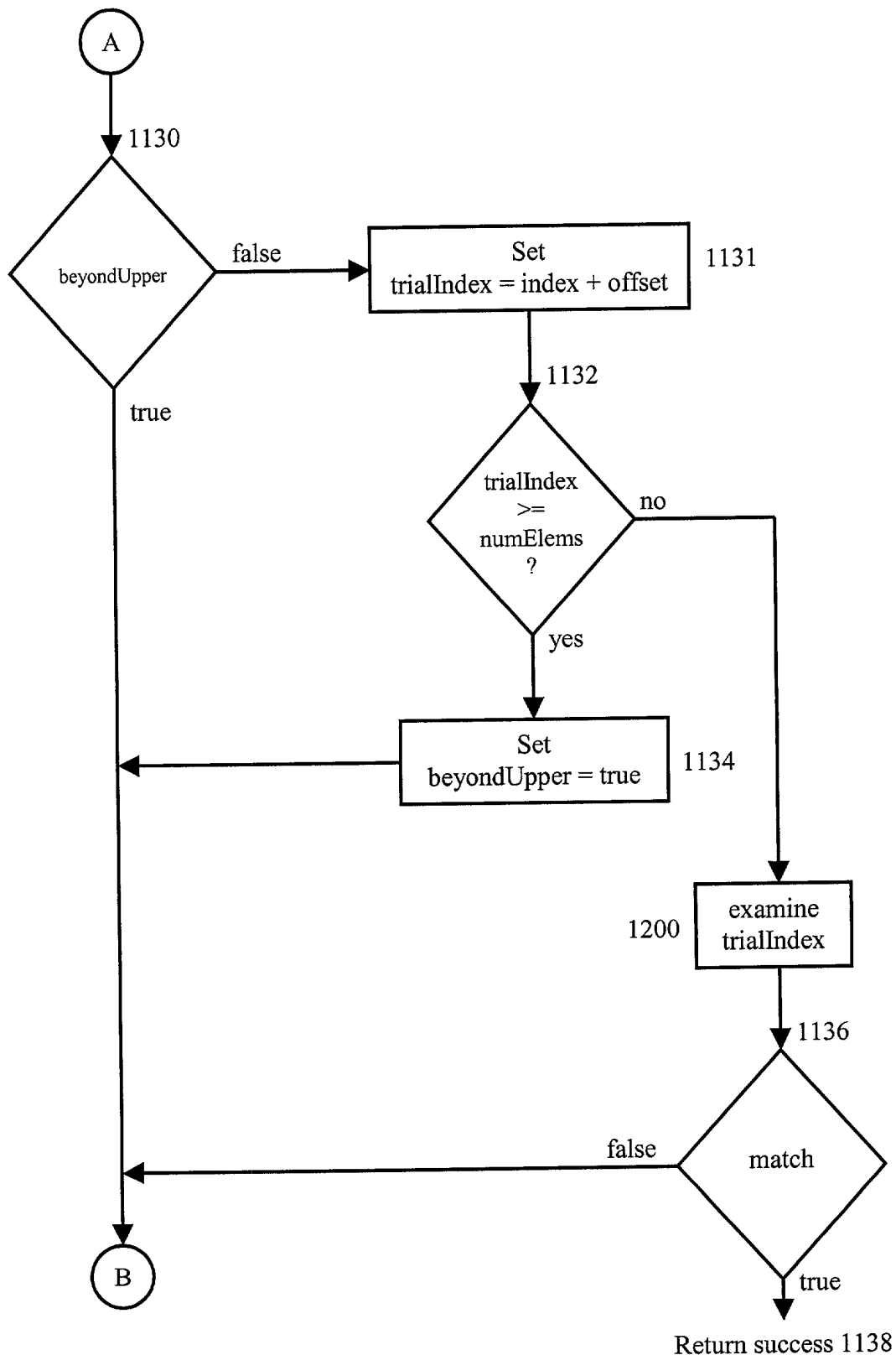


Figure 11 (Sheet 2 of 3)

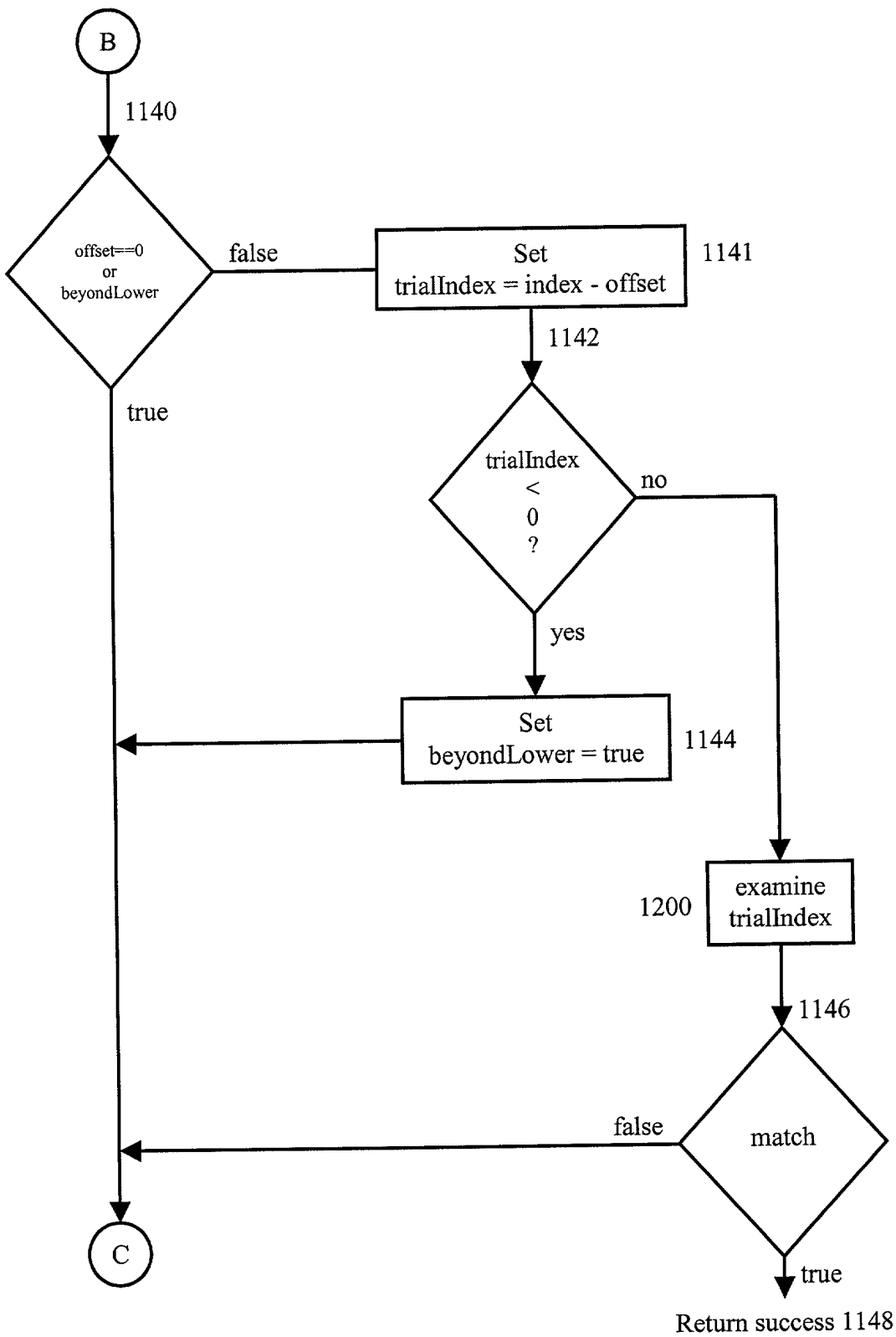
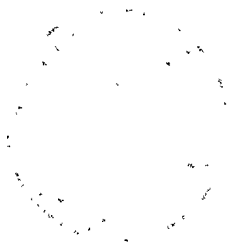
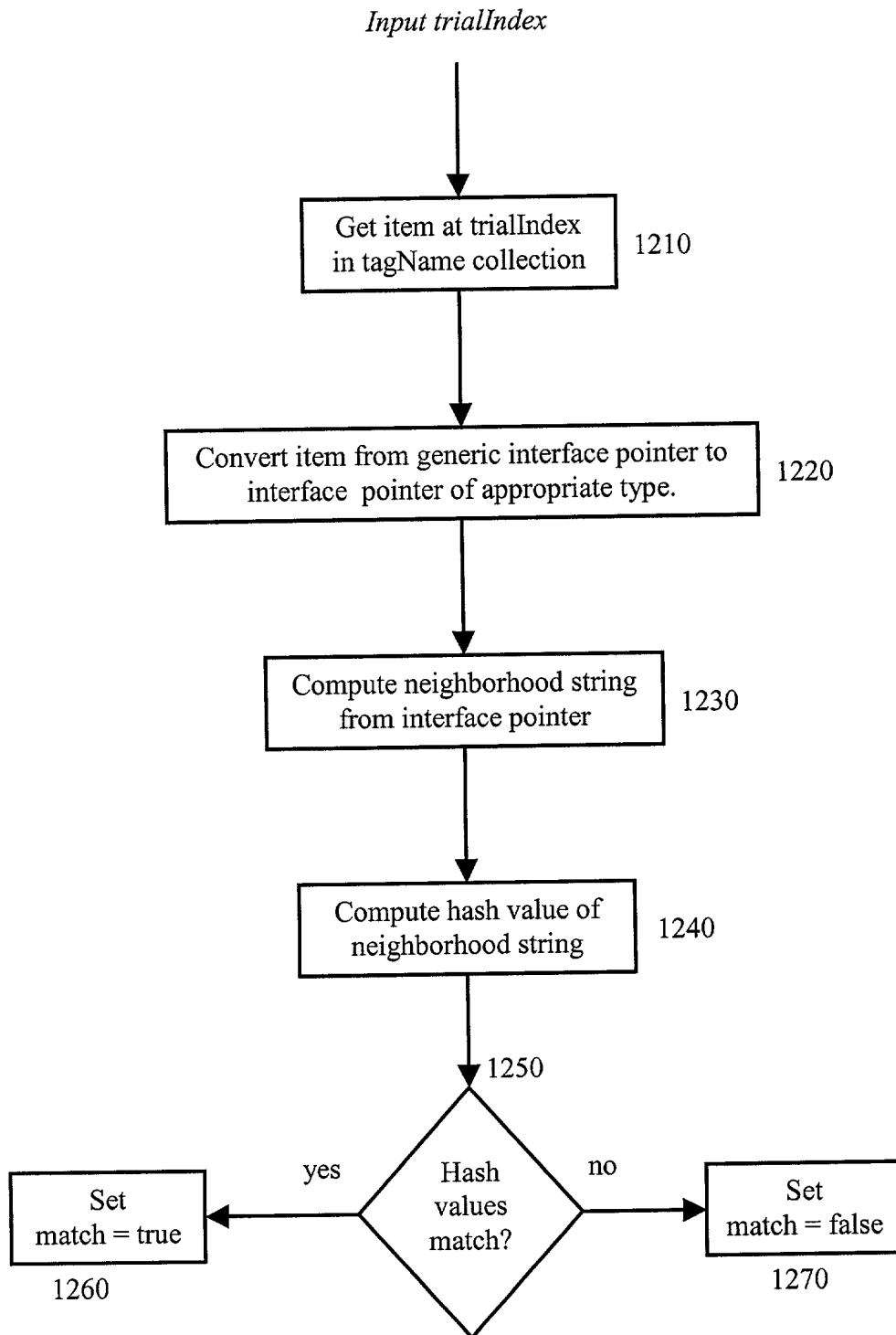
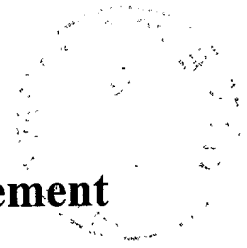


Figure 11 (Sheet 3 of 3)

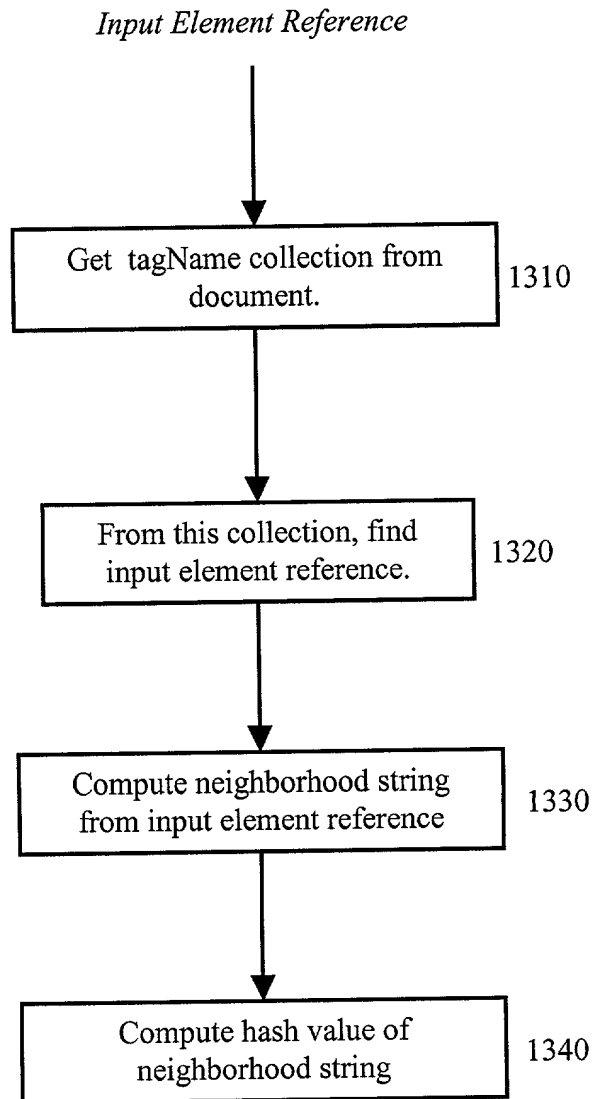
Figure 12 – Examine trialIndex





**Figure 13 – Compute Element Locator from Element Reference**

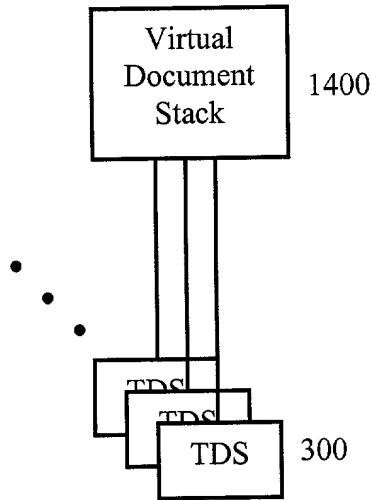
1300



Approved for Release by NSA on 09-11-2013 pursuant to E.O. 13526



**Figure 14 – Virtual Document Stack**



09/944,062